Applied Text Analytics & Natural Language Processing

with Dr. Mahdi Roozbahani
& Wafa Louhichi

Transformers
Introduction – Part 2

Structure of the Transformer Model

This is the overall structure of the Transformer model.

- **Encoder**: gets the input and creates a representation of it.
- **Decoder**: gets the input representation along with the target output and generates the new output (outputs are generated one word at a time and is fed back to the decoder)

Representation: a set of vectors produced by the encoder that can be read by the decoder to acquire information about the input. Embeddings of similar words are similar to each other.

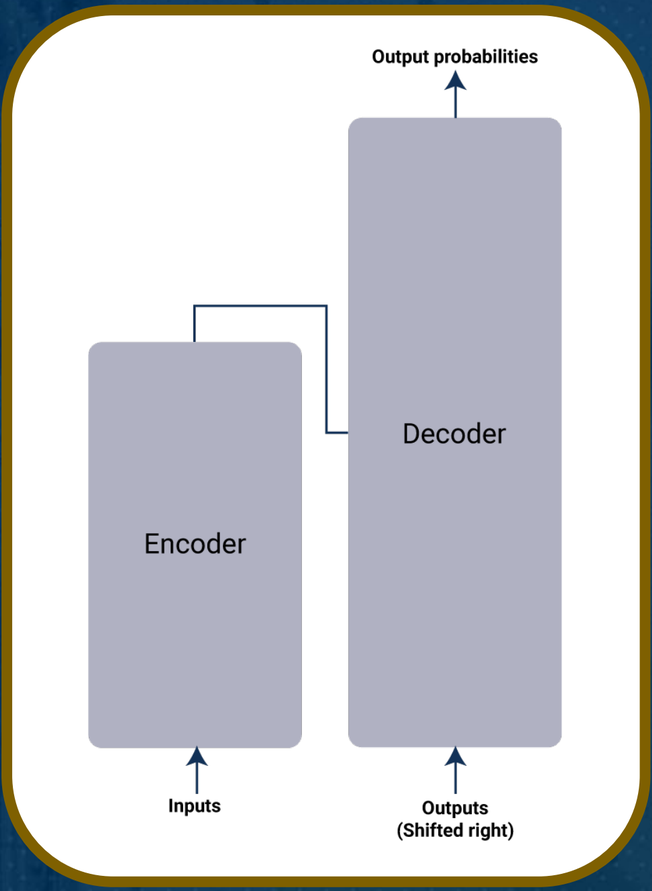


Image from Hugging Face course

Encoder and Decoder Can Be Used Separately in Structures:

Encoder-only models: for tasks that require understanding the input. *E.g., what is the meaning of this sentence?, Fill in the blank with the right word.*

- Good in the understanding of input, extracting meaningful information, sequence classification, masked language modeling, and natural language understanding (NLU)
- BERT, RoBERTa, ALBERT
- Bi-directional

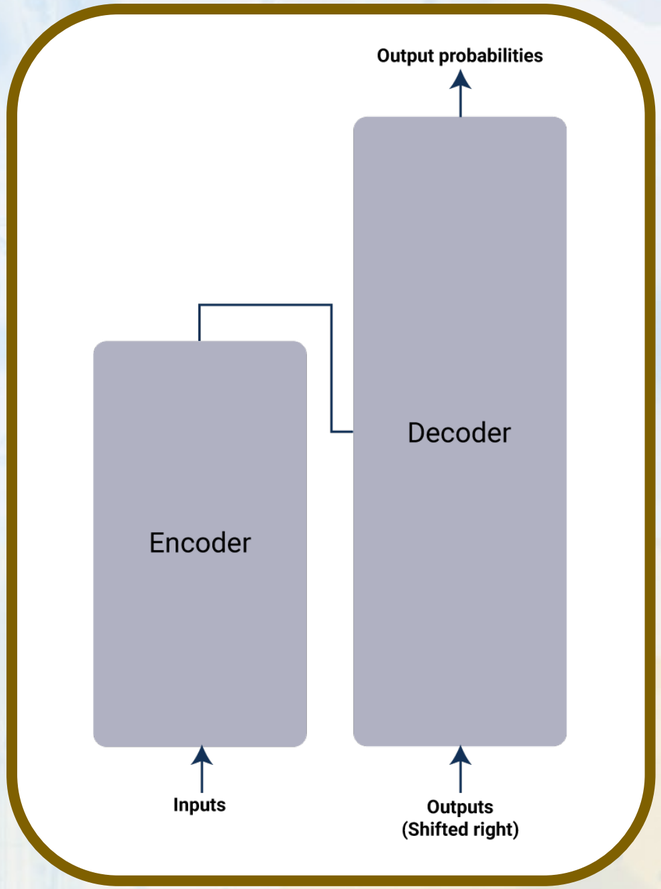


Image from Hugging Face course

https://huggingface.co/course/chapter1/1

BERT Neural Network - EXPLAINED! (https://www.youtube.com/watch?v=xI0HHN5XKDo)

Encoder and Decoder Can Be Used Separately in Structures:

Decoder-only models: for generative tasks. *E.g., Guessing the next word in a sentence.*

- Good in generating sequences, natural language generation (NLG)
- GPT, GPT-2, GPT-3
- Uni-directional

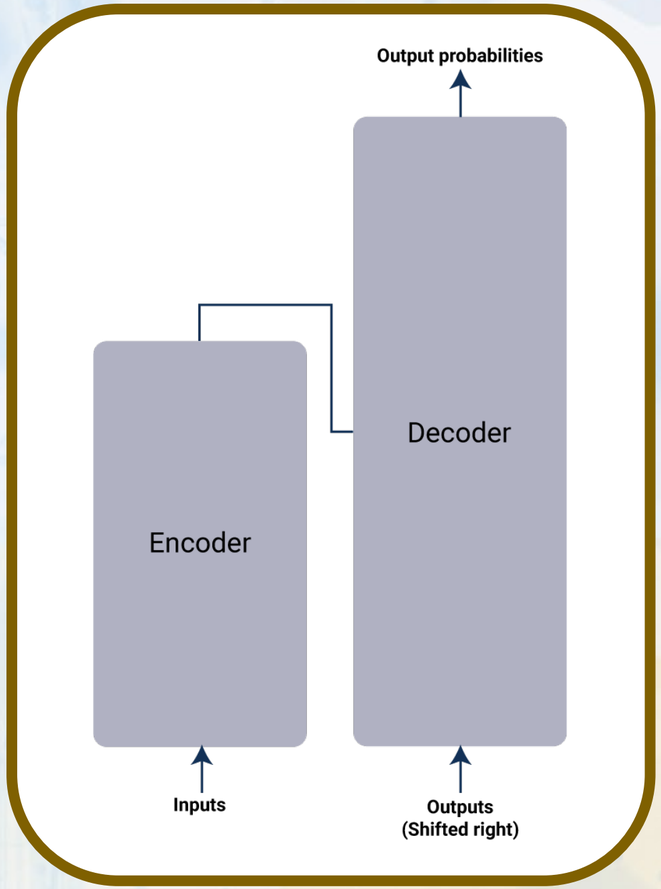


Image from Hugging Face course

https://huggingface.co/course/chapter1/1

BERT Neural Network - EXPLAINED! (https://www.youtube.com/watch?v=xI0HHN5XKDo)

Encoder and Decoder Models (Sequence-To-Sequence)

Encoder-decoder models: for generative tasks that require input. They are good for seq2seq tasks. *E.g., Translation, text summarization*

- BART and T5 use this structure

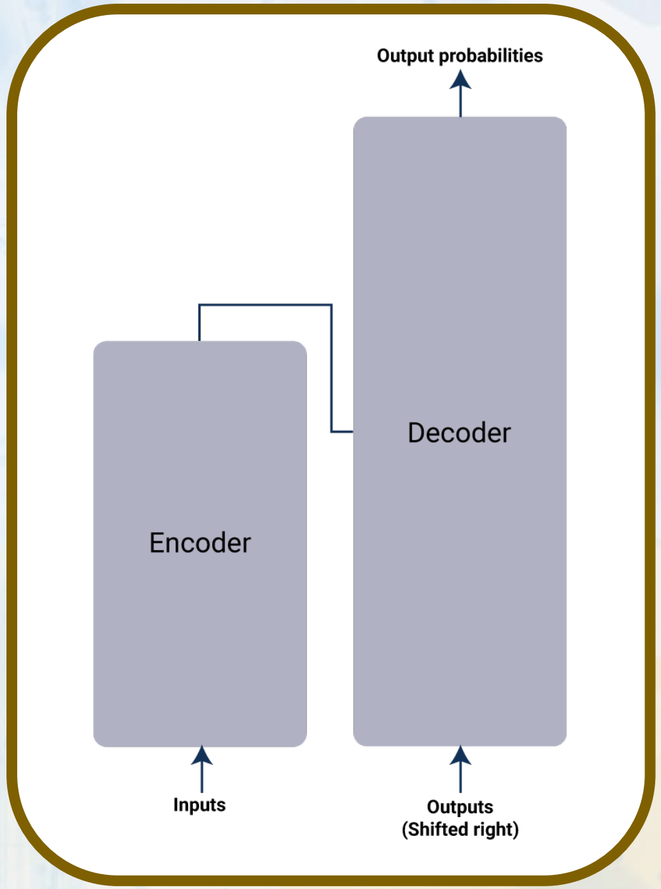


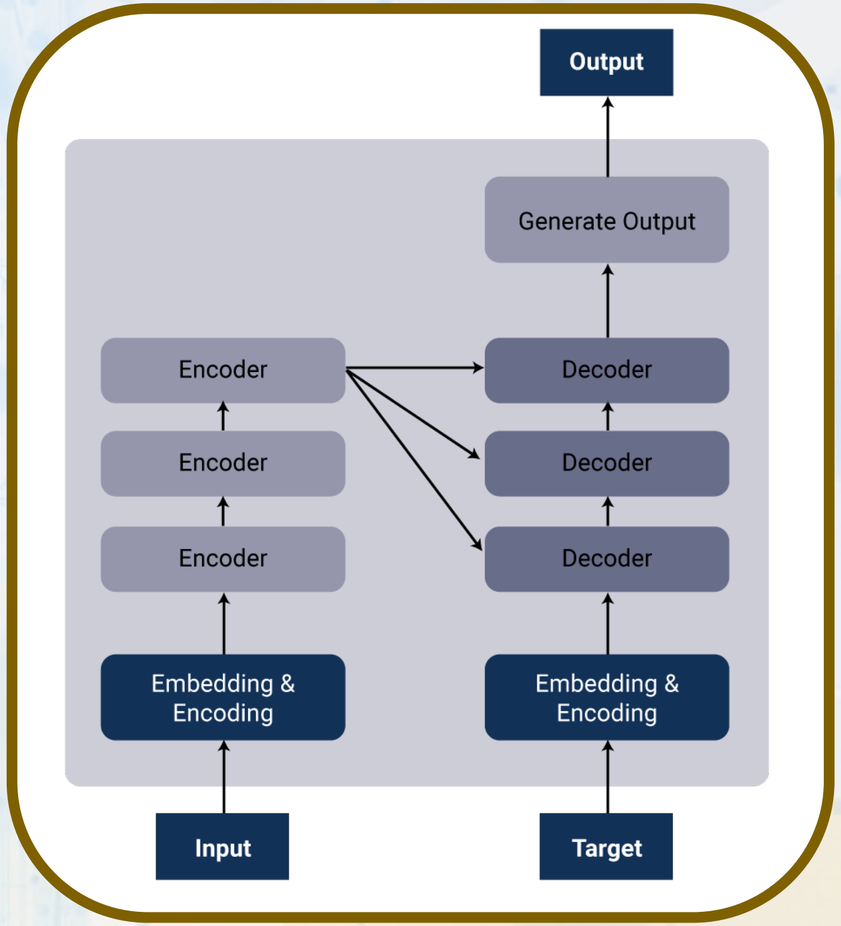
Image from Hugging Face course

https://huggingface.co/course/chapter1/1

BERT Neural Network - EXPLAINED! (https://www.youtube.com/watch?v=xI0HHN5XKDo)

Transformer Architecture Encoder Stack

- It contains encoder blocks (usually 6, here 3)
- Receives its input from Embedding and Position Encoding
- Encoder duty: mapping its input to a fixed-length vector, called a context vector
  - It contains a crucial component = Self-Attention
  - The context vector is fed to the decoder.

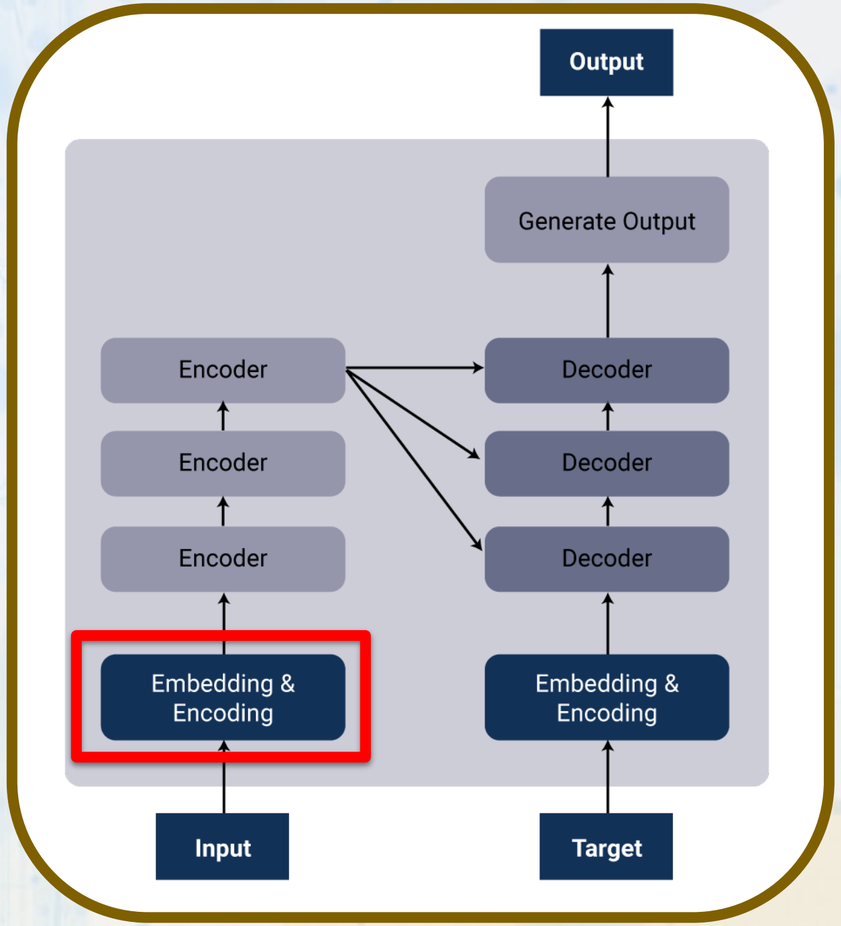


Embedding + Positional Encoding

- Embeddings come from pre-trained models
  - Pre-trained models such as Word2Vec (Google), GloVe (Stanford), …
- Ordering words is important to grasp the meaning of a sentence
- Encoding (Positional Encoding) encodes the position of each word. This information is added to the embedding result and is fed to the encoder

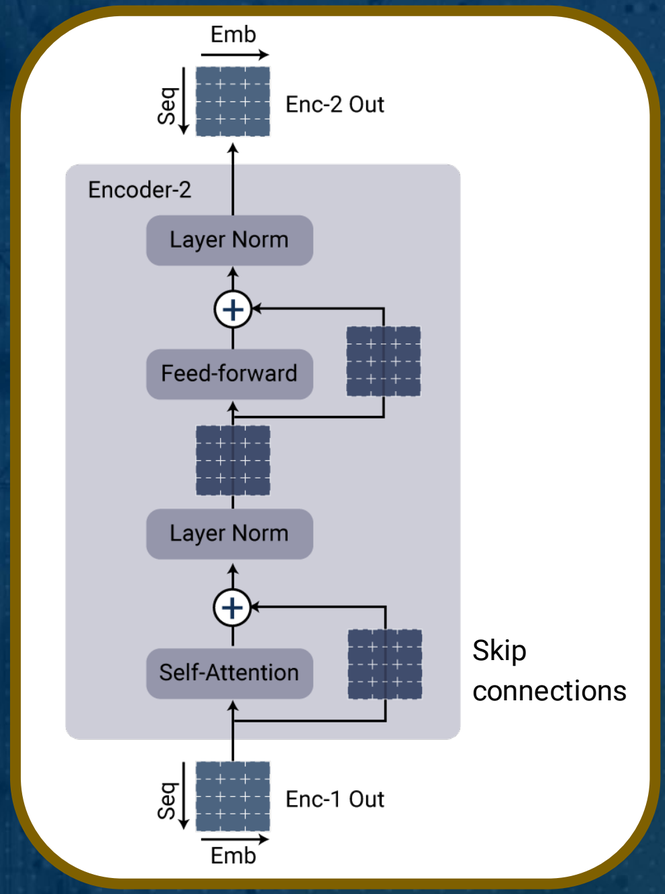
PE: positional Encoding
Pos: position of the word
d: length of encoding vector (i.e. word2vec)

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}})$$
$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}})$$



Encoder Block

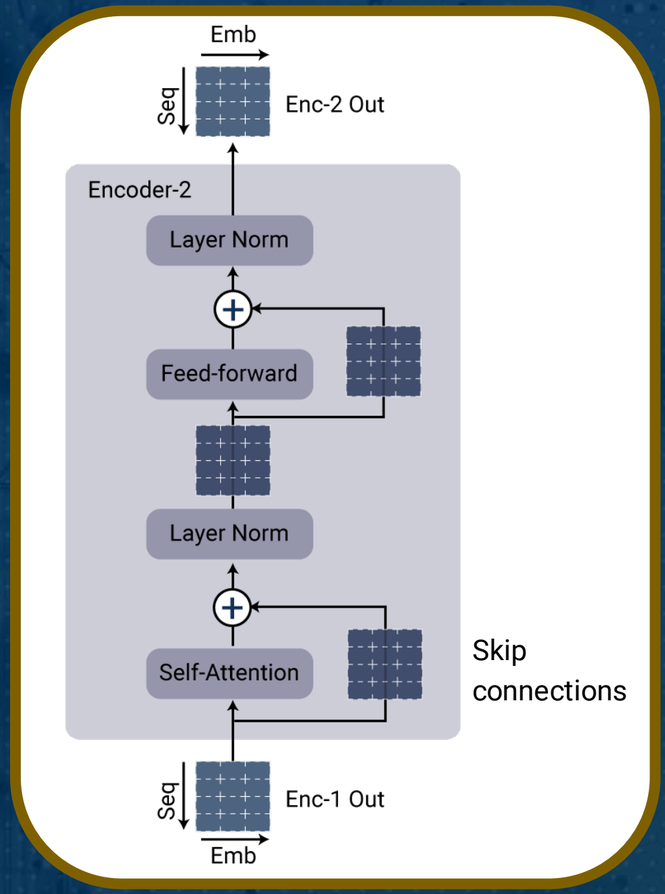
- **Self-attention layer**
  - Computes the relationship between words in the input sequence
- **Skip connections**
  - Helps with exploding/vanishing gradient problems
- **Normalization layers**
  - Provides smoother gradients, faster training, and better generalization accuracy by normalizing the distribution of intermediate layers
- **Feed-forward (linear) layer**
  - Consists of two linear layers with ReLU activation between them. Its main purpose is to transform the attention vectors into a form that is acceptable by the next encoder or decoder layer
  - Each feed-forward layer has its own parameters:
    - $FFN(x) = ReLu(x\theta_1 + b_1) * \theta_2 + b_2$



Xu, Jingjing, et al. "Understanding and improving layer normalization." Advances in Neural Information Processing Systems 32 (2019).
Geva, Mor, et al. "Transformer feed-forward layers are key-value memories." arXiv preprint arXiv:2012.14913 (2020).

Encoder Block

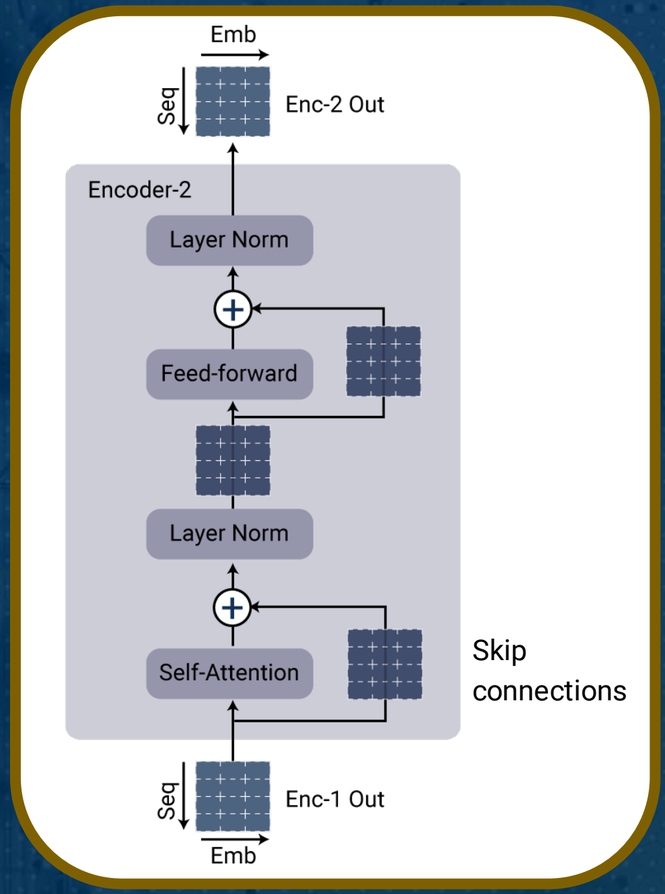
- **Self-attention layer**
  - Computes the relationship between words in the input sequence
- **Skip connections**
  - Helps with exploding/vanishing gradient problems
- **Normalization layers**
  - Provides smoother gradients, faster training, and better generalization accuracy by normalizing the distribution of intermediate layers
- **Feed-forward (linear) layer**
  - Consists of two linear layers with ReLU activation between them. Its main purpose is to transform the attention vectors into a form that is acceptable by the next encoder or decoder layer
  - Each feed-forward layer has its own parameters:
    - $FFN(x) = ReLu(x\theta_1 + b_1) * \theta_2 + b_2$



Xu, Jingjing, et al. "Understanding and improving layer normalization." Advances in Neural Information Processing Systems 32 (2019).
Geva, Mor, et al. "Transformer feed-forward layers are key-value memories." arXiv preprint arXiv:2012.14913 (2020).

Encoder Block

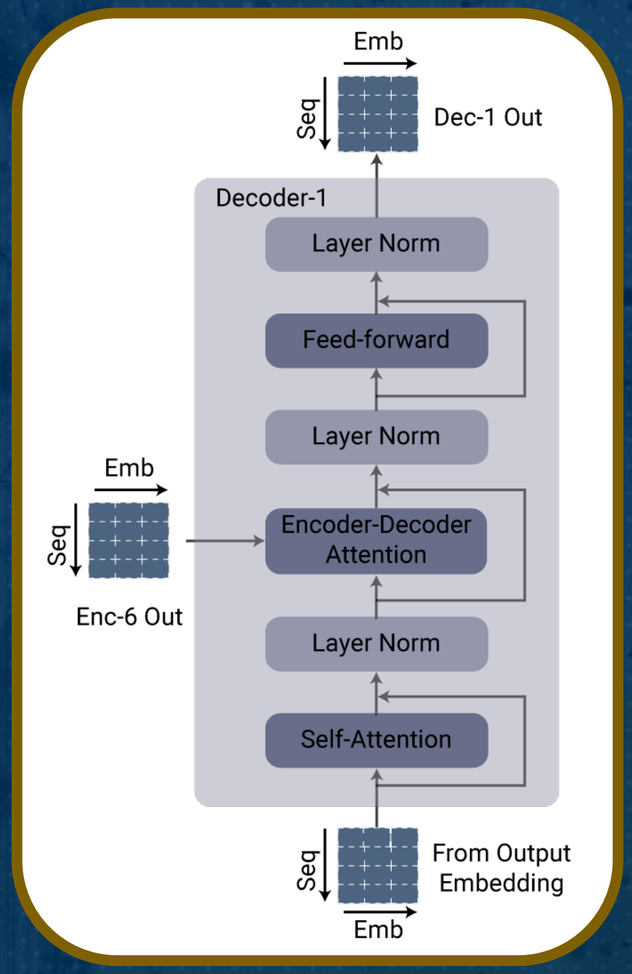
- **Self-attention layer**
  - Computes the relationship between words in the input sequence
- **Skip connections**
  - Helps with exploding/vanishing gradient problems
- **Normalization layers**
  - Provides smoother gradients, faster training, and better generalization accuracy by normalizing the distribution of intermediate layers
- **Feed-forward (linear) layer**
  - Consists of two linear layers with ReLU activation between them. Its main purpose is to transform the attention vectors into a form that is acceptable by the next encoder or decoder layer
  - Each feed-forward layer has its own parameters:
    - $FFN(x) = ReLu(x\theta_1 + b_1) * \theta_2 + b_2$



Xu, Jingjing, et al. "Understanding and improving layer normalization." Advances in Neural Information Processing Systems 32 (2019).
Geva, Mor, et al. "Transformer feed-forward layers are key-value memories." arXiv preprint arXiv:2012.14913 (2020).

Decoder Block

- **Self-attention layer**
  - Computes the relationship between words in the target sequence
- **Encoder-decoder attention layer**
- **Normalization layer**
- **Feed-forward (linear) layer**



Xu, Jingjing, et al. "Understanding and improving layer normalization." Advances in Neural Information Processing Systems 32 (2019).
Geva, Mor, et al. "Transformer feed-forward layers are key-value memories." arXiv preprint arXiv:2012.14913 (2020).

Original Transformer Architecture

- The attention layer in the encoder has access to the whole input sequence since it needs to pay attention to other words (before and after) when creating a representation for a specific word
- The bottom attention layer in the decoder is a self-attention unit and has access to only the words translated so far. The top attention layer is a regular attention unit and uses the output of the encoder to produce the most accurate output
- The Transformer model is "auto-regressive"; it receives the recently generated output as the new input for prediction

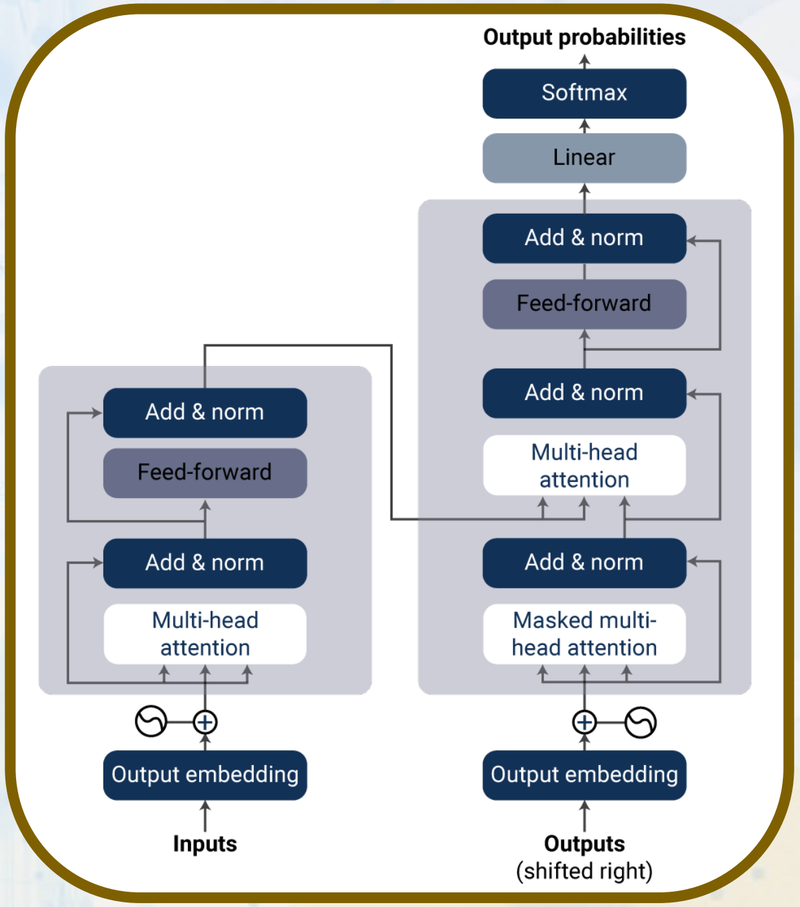


Image from Hugging Face course

https://huggingface.co/course/chapter1/1
Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).