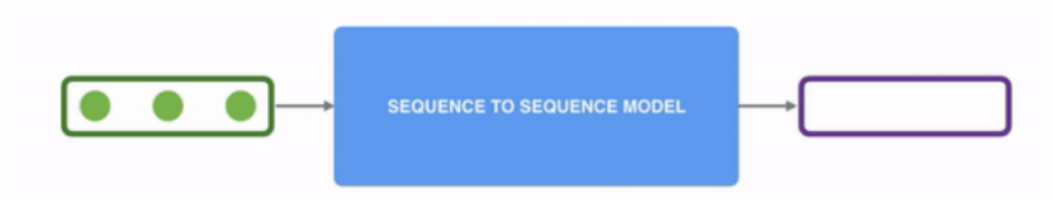
10: Transformers (Encoder-Decoder Architectures)

Seq2Seq Learning algorithms

These are algorithms where inputs and outputs are a set of sequences. These sequences can be anything like words, characters, etc.

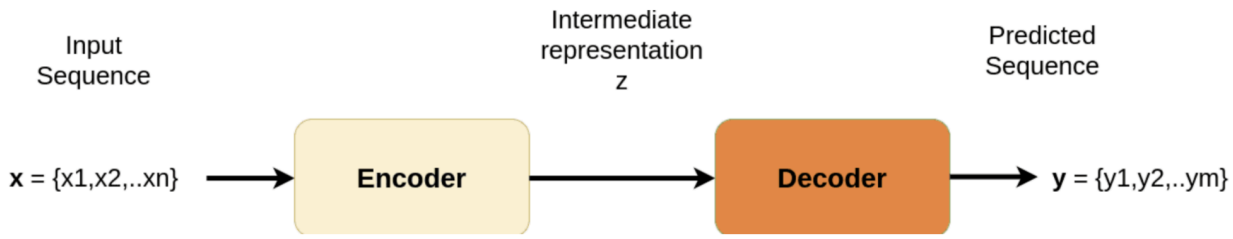


Machine translation is applying seq2seq algorithms to translate sentences from one language to another. Major algorithms for machine translation include

Abbreviation	RBMT	SMT	NMT
Name	Rule Based Machine Translation	Statistical Machine Translation	Neural Machine Translation
Overview	Machine translation based on dictionary and knowledge of grammar	Machine translation based on statistical information derived from large volume of collected parallel data	Utilizes AI Machine translation using deep learning technology
Pros	Translation speed is fast Faithful to the original Suitable for standard translation	Translations are relatively natural sentences, higher than the quality of RBMT It is also easy to get high BLEU* scores with SMT	Can produce more natural sentences, and higher translation quality than SMT
Cons	Translation lacks naturalness	Translation lacks naturalness	Takes time to learn target data

Neural Machine Translation is the state-of-the-art algorithm as of 2023

Encoder-Decoder Architecture

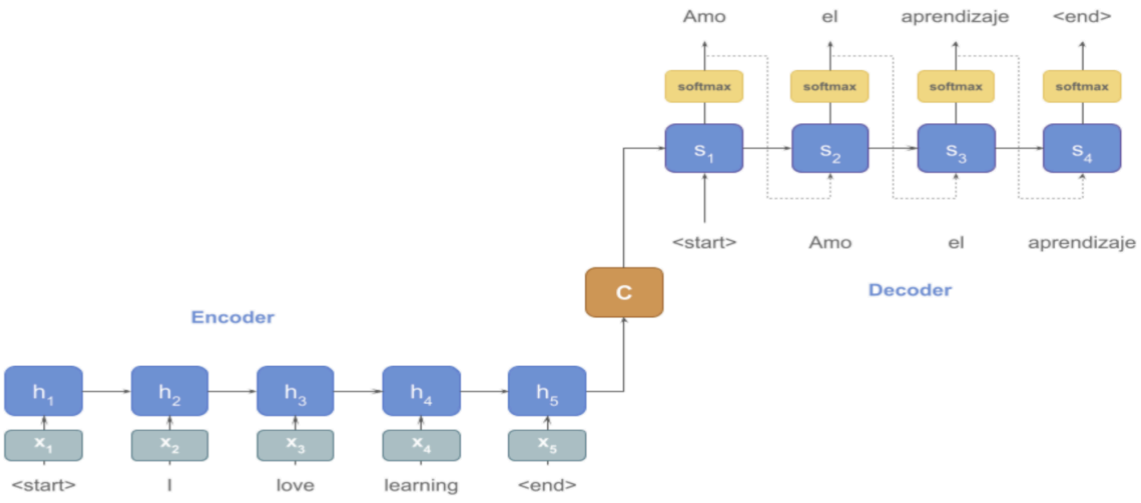


Encoder decoder architecture consists of two components

- The encoder processes the input sequence to a hidden representation, which captures the inner meaning of the text in terms of hidden vectors
- The decoder converts hidden representation to output sequence

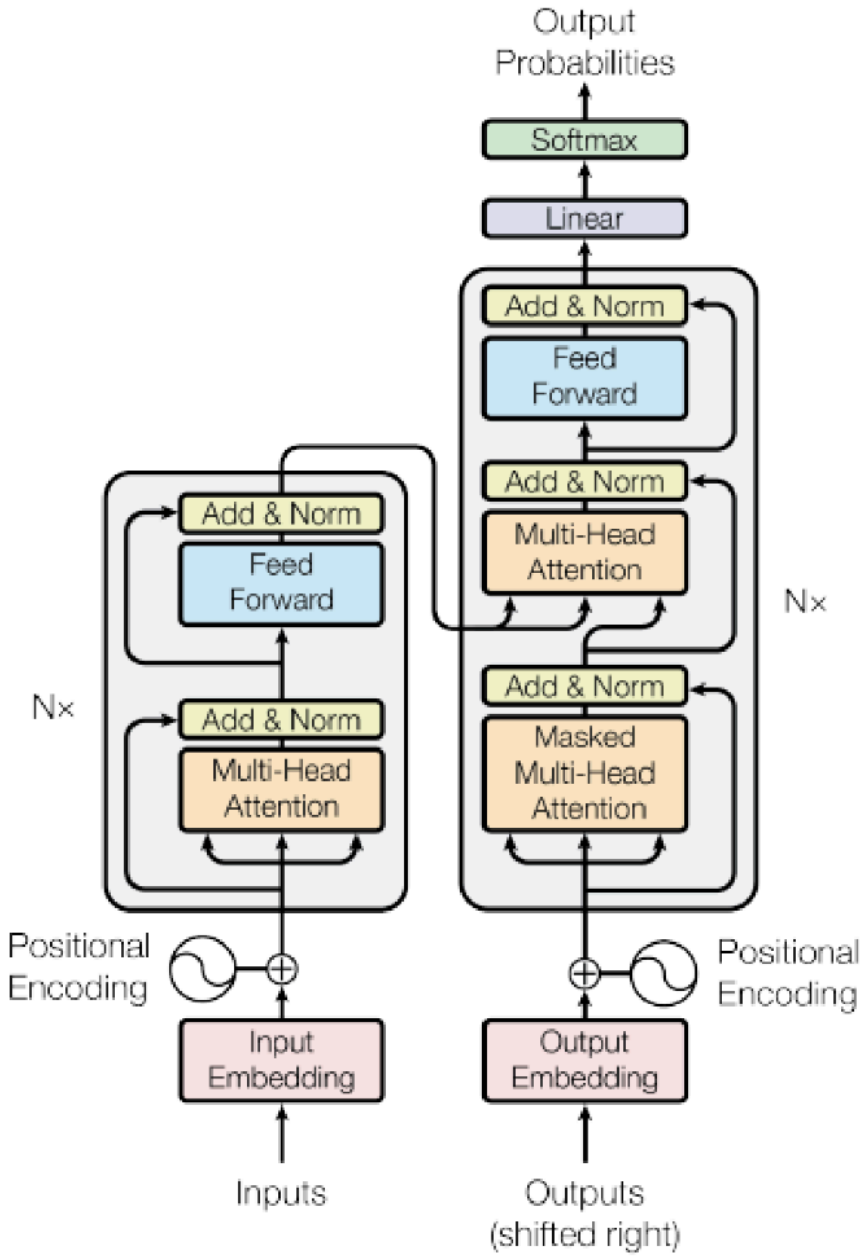
RNN/LSTM/GRU based Encoder-Decoder

- Encoder
  - Receives token embeddings as input
  - Processed through multiple RNN stacks
  - The last hidden state (and cell state) captures hidden representation in the RNN/LSTM/GRU based encoder
- Decoder
  - Initialized with the encoder's hidden representation
  - Processed through multiple RNN stacks
  - Outputs from the final RNN layer are used to calculate loss/logits



- Problems with RNN/LSTM/GRU based NMT
  - Unable to capture long-term dependencies
  - Architecture is not parallelizable

Transformers based Encoder-Decoder



- Encoder block
  - Embeddings of tokenized inputs are added to positional embeddings to serve as inputs to the encoder block
  - Each encoder block contains a Multi-Head Attention network and a Feedforward network. Outputs of these networks are added to it's inputs and then normalized.
  - Each head of Multi-Head Attention computes attention patterns between projected queries and keys. This pattern is a matrix multiplied by values. Multiplined values from each head are concatenated and projected to get outputs.
  - Feed-forward network is a simple linear projection to a hidden dimension and another projection to embed dimension.
- Decoder block
  - Embeddings of tokenized inputs are added to positional embeddings to serve as inputs to the decoder block too.
    - During training, inputs to a decoder block are outputs shifted to the right.
    - During inference, inputs to a decoder block are usually just a start token
  - Each decoder block contains 4 components: Masked multi-head attention, feedforward 1, multihead attention, and feedforward 2.
  - In masked multi-head attention, the upper triangular mask is applied to the attention pattern before matmul by values. Other components and flow inputs are similar to that of the encoder.
  - Finally, outputs are linear projected to the vocab dimension and logits are obtained. Categorical cross entropy is typically used as loss function.