Encoder-Decoder Model

There are three main blocks in the encoder-decoder model,

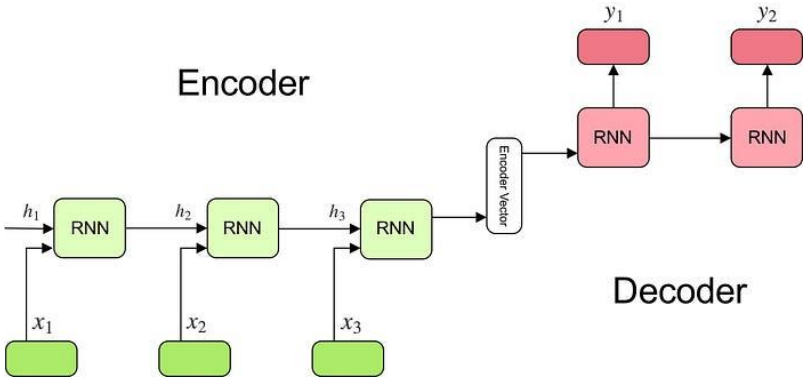
- Encoder
- Hidden Vector
- Decoder

The Encoder will convert the input sequence into a single-dimensional vector (hidden vector). The decoder will convert the hidden vector into the output sequence.

Encoder-Decoder models are jointly trained to maximize the conditional probabilities of the target sequence given the input sequence.

How the Sequence to Sequence Model works?

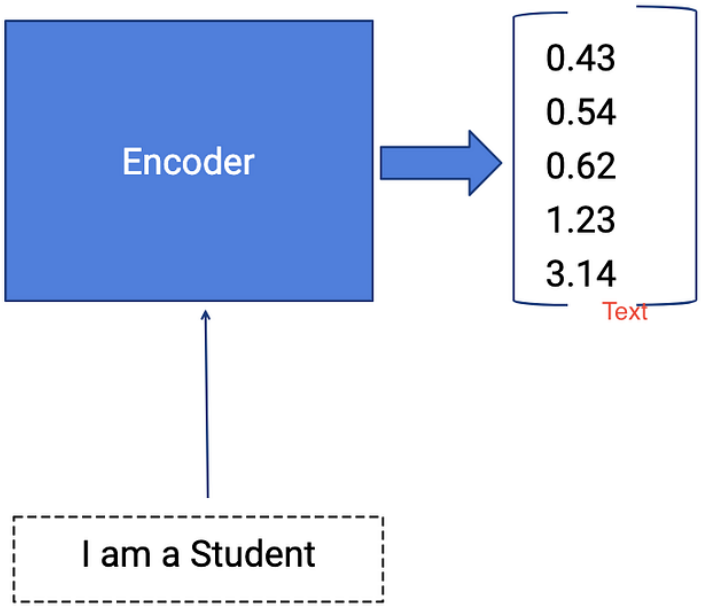
In order to fully understand the model's underlying logic, we will go over the below illustration:



Encoder-decoder sequence to sequence model

Encoder

- Multiple RNN cells can be stacked together to form the encoder. RNN reads each inputs sequentially
- For every timestep (each input) t, the hidden state (hidden vector) h is updated according to the input at that timestep X[i].
- After all the inputs are read by encoder model, the final hidden state of the model represents the context/summary of the whole input sequence.
- Example: Consider the input sequence “I am a Student” to be encoded. There will be totally 4 timesteps (4 tokens) for the Encoder model. At each time step, the hidden state h will be updated using the previous hidden state and the current input.



Example: Encoder

- At the first timestep t1, the previous hidden state h0 will be considered as zero or randomly chosen. So the first RNN cell will update the current hidden state with the first input and h0. Each layer outputs two things — updated hidden state and the output for each stage. The outputs at each stage are rejected and only the hidden states will be propagated to the next layer.

- The hidden states *h_i* are computed using the formula:

$$h_t = f(W^{(hh)} h_{t-1} + W^{(hx)} x_t)$$

- At second timestep t2, the hidden state h1 and the second input X[2] will be given as input , and the hidden state h2 will be updated according to both inputs. Then the hidden state h1 will be updated with the new input and will produce the hidden state h2. This happens for all the four stages wrt example taken.

- A stack of several recurrent units (LSTM or GRU cells for better performance) where each accepts a single element of the input sequence, collects information for that element, and propagates it forward.

- In the question-answering problem, the input sequence is a collection of all words from the question. Each word is represented as *x_i* where *i* is the order of that word.