Word2vec embedding - apple - embedding - same

Keras embedding - apple - same - update based on the context - bidirectional - this may end up in understanding the context -

**LSTM** - Long Short Term Memory

Successor - RNN(Recurrent Neural Network)

Input - One by one Token is passed

Output - Token

Advantage - Overcome Exploding and vanishing gradient Descent. It holds

memory(context)

Disadvantage - Cannot remember longer context

Application - Time Series, Next Token prediction (WhatsApp, Gmail), Sentiment Analysis, Text Classification

Sequence to Sequence Model (Seq2Seq): Encoder Decoder Model

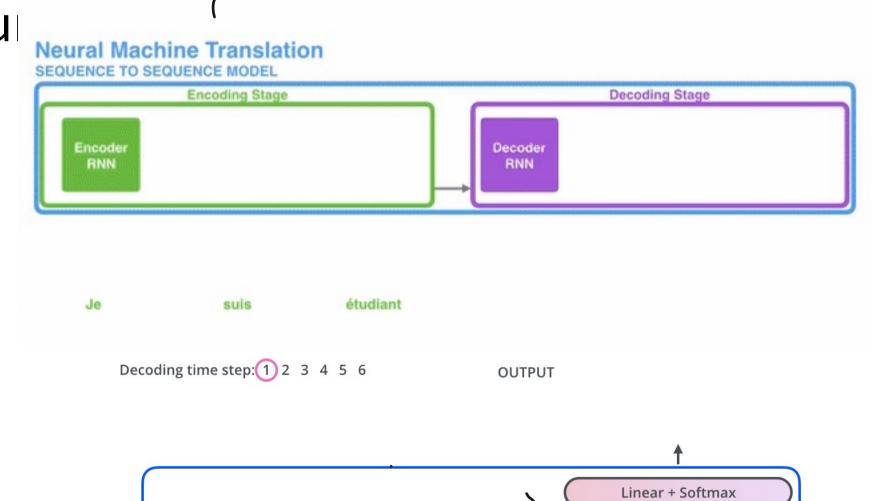
If the input and output is of different length

Encoder - LSTM Decoder - LSTM

Advantage - Complex(anything) Sequential data(input can be any thing of any size)

Disadvantage Longer context is different, Computational complexity, one by one token input, the decoder will get the final hidden input from the encoder after getting all the data inputs, same word can have different meaning in different context

Application - Machine Translation, Text Su Answering



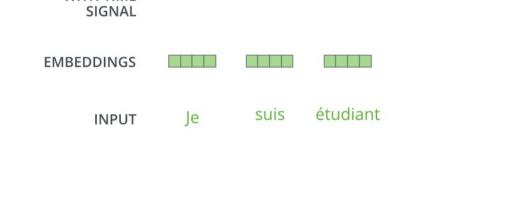
### TRANSFORMER

#### Advantage:

- 1. Encoder can take the whole token at once
- 2. It can understand long context
- 3. The same word with different meaning based on the context is EMBEDDING WITH TIME SIGNAL

# Disadvantage:

- 1. Computational cost, time, resource, data
- 2. No Interpretibilty
- 3. Heavy carbon footprint



# Attention Mechanism - Transformer

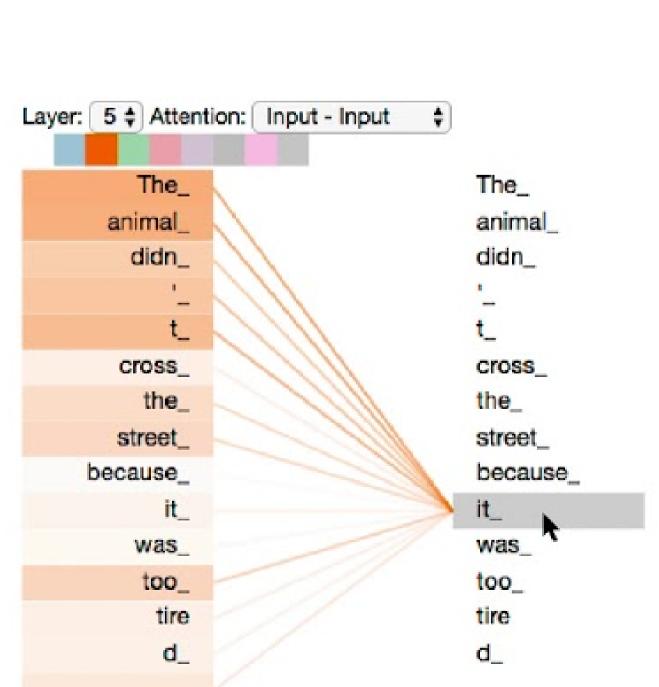
#### 2017 Google Brain

Importance of updating all the embedding of token based on the other tokens inside the sentence (Self Atten

# Encoder -

Self Attention - Muti Head Attention - Encoder - Several Encoders

# Self-Attention



- Consider two input sentences we want to translate:
  - The animal didn't cross the street because it was too tired
  - The animal didn't cross the street because it was too wide
- "it" refers to "animal" in first case, but to "street" in second case; this is hard for traditional Seq2Seq models to model
- As the model processes each word, self-attention allows it to look at other positions in input sequence to help get a better encoding
- Recall RNNs: we now no longer need to maintain a hidden state to incorporate representation of previous words/vectors!

Encoders - multiple - 6 Encoders

Vineeth N B (IIT-H)

Each Encoder - multiple Attention - 12 to many

Each Attention - update the Embedding



§9.5 Self-Attention and Transformers