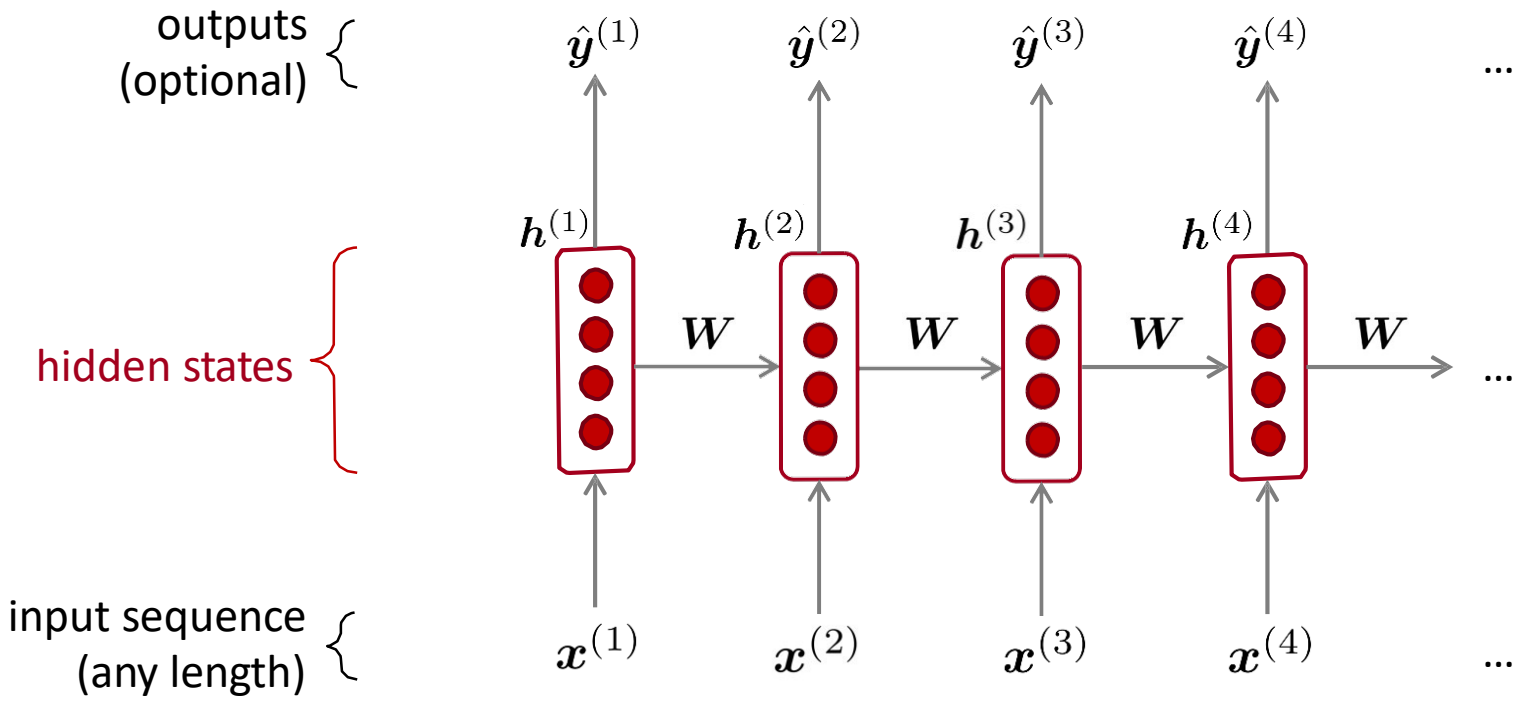
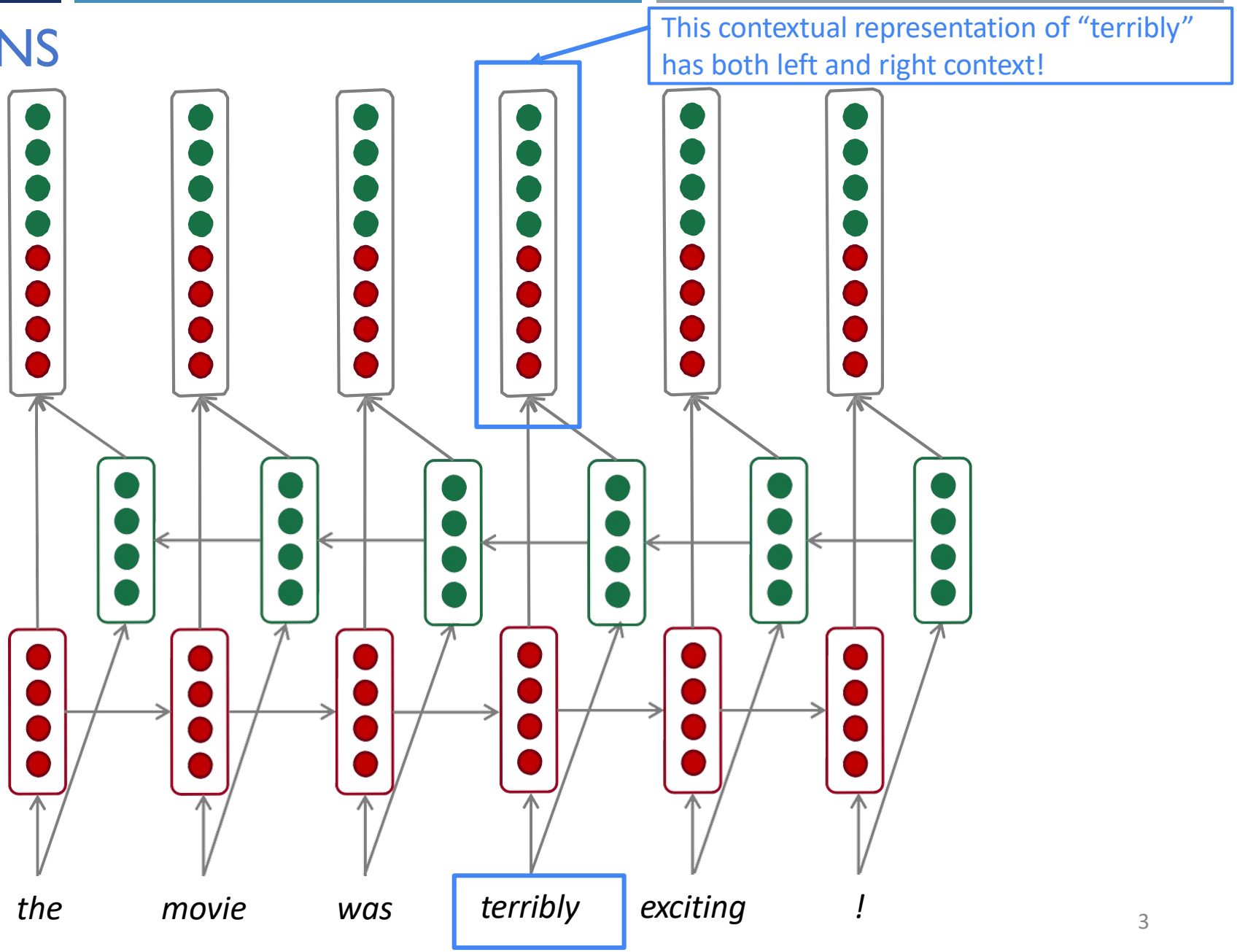
Lecture 08:
Encoder - decoder model

RECURRENT NEURAL NETWORK

- RNN apply the same weights (W) repeatedly
- Hidden state $h^{(t)}$ depends on the output of the previous state $h^{(t-1)}$, $h^{(t-1)}$ is a variant of $h^{(t)}$.



BIDIRECTIONAL RNNS



Credits: Chris Manning

BIDIRECTIONAL RNNS

On timestep *t*:

This is a general notation to mean "compute one forward step of the RNN" – it could be a vanilla, LSTM or GRU computation.

Forward RNN $\overrightarrow{\boldsymbol{h}}^{(t)} = \mathrm{RNN}_{\mathrm{FW}}(\overrightarrow{\boldsymbol{h}}^{(t-1)}, \boldsymbol{x}^{(t)})$

Backward RNN $\overleftarrow{\boldsymbol{h}}^{(t)} = \mathrm{RNN}_{\mathrm{BW}}(\overleftarrow{\boldsymbol{h}}^{(t+1)}, \boldsymbol{x}^{(t)})$

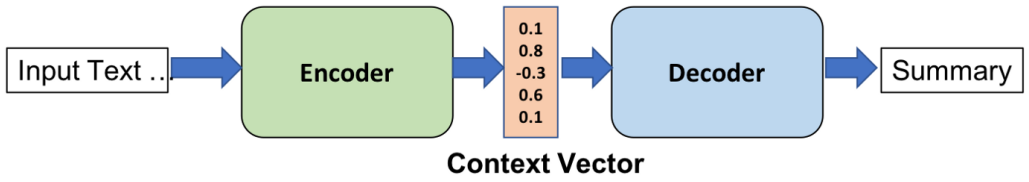
Generally, these two RNNs have separate weights

Concatenated hidden states $\boldsymbol{h}^{(t)} = [\overrightarrow{\boldsymbol{h}}^{(t)}; \overleftarrow{\boldsymbol{h}}^{(t)}]$

We regard this as "the hidden state" of a bidirectional RNN. This is what we pass on to the next parts of the network.

Credits: Chris Manning

ENCODER – DECODER ARCHITECTURE



Encoder:

- at each time step take a single input of entire sequence.
- process the entire sequence and output a context vector

Context vector:

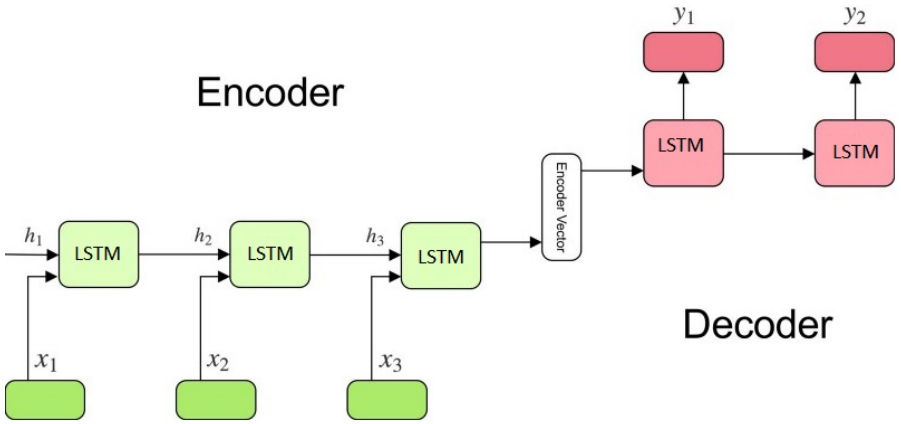
- conveys the essence of the input to the decoder.

Decoder:

- initialized from the final states of the Encoder (context vector).
- using initial states, decoder generates the output sequence.

ENCODER – DECODER ARCHITECTURE

Let us consider a vocabulary *V*, an encoder-decoder maps a sequence $x = (x_1, \dots, xn)$ onto another sequence $y = (y_1, \dots, ym)$, where all $x_i, yj \in V$



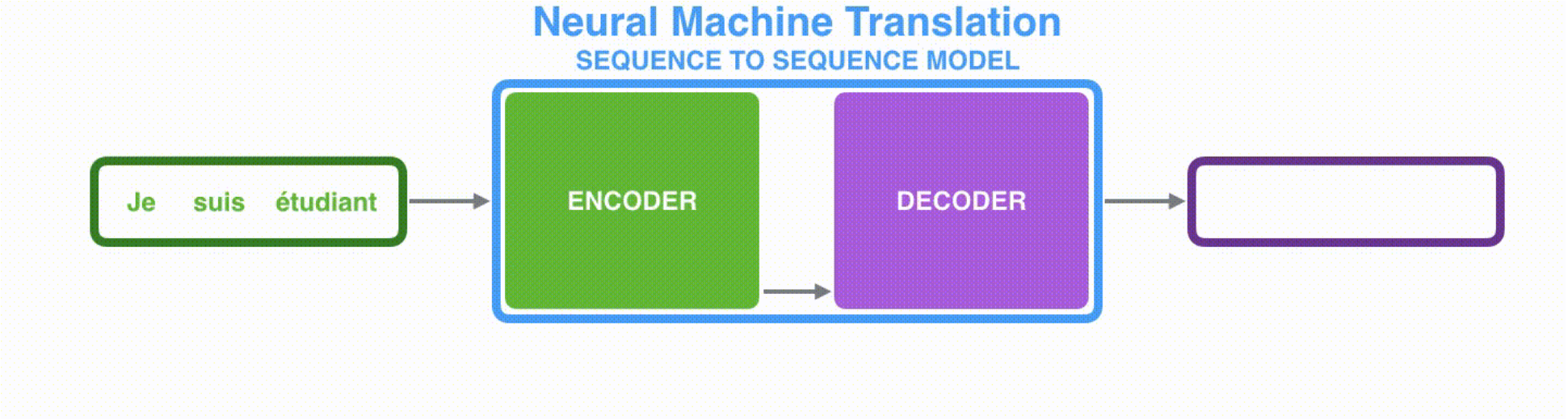
$$\mathbf{h}_t = g(\mathbf{h}_{t-1}, \mathbf{x}_t)$$
$$\mathbf{y}_t = f(\mathbf{h}_t)$$

- at time t the output y_t and hidden state h_t are computed as

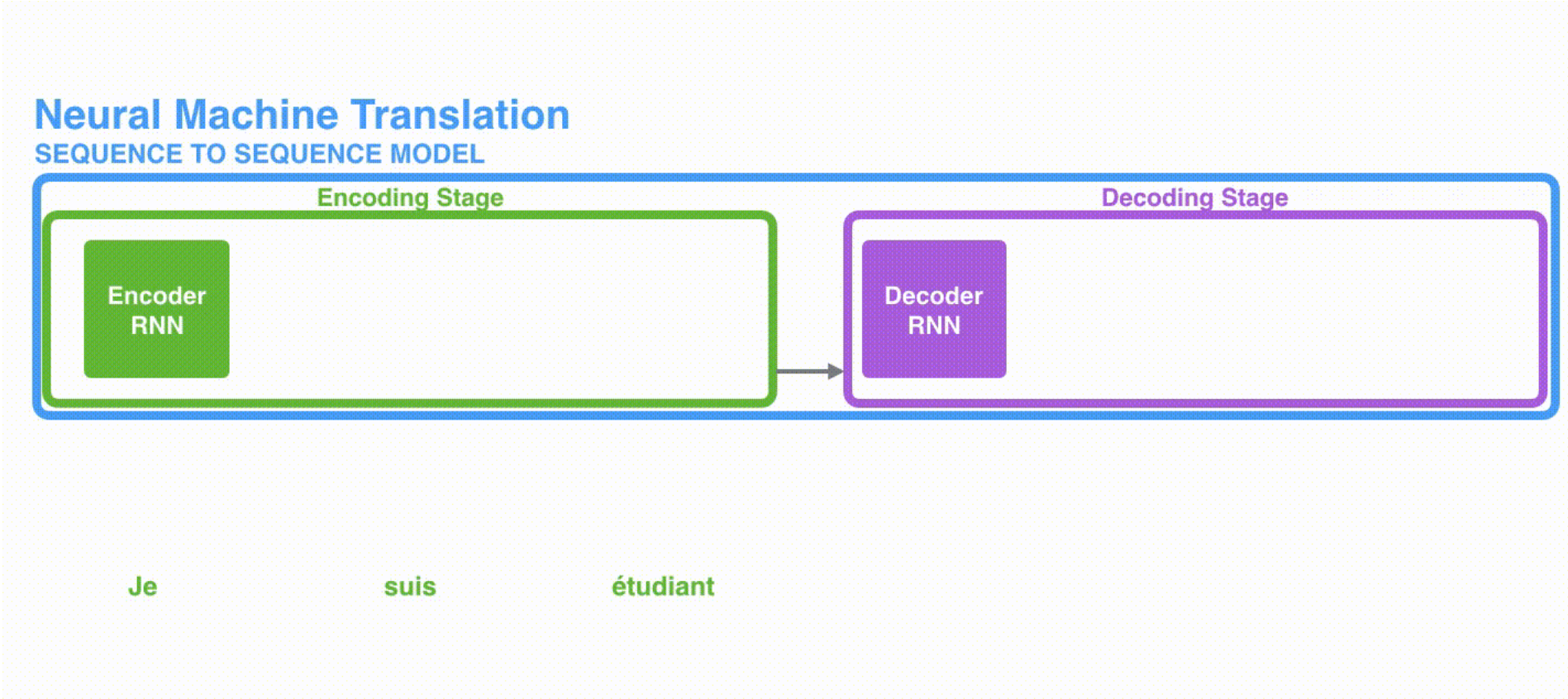
ENCODER – DECODER (SEQ2SEQ) MODEL

The model is composed of an encoder and a decoder.

- The encoder processes each item in the input sequence, it compiles the information it captures into a vector (called the context).
- After processing the entire input sequence, the encoder sends the context over to the decoder, which begins producing the output sequence item by item.



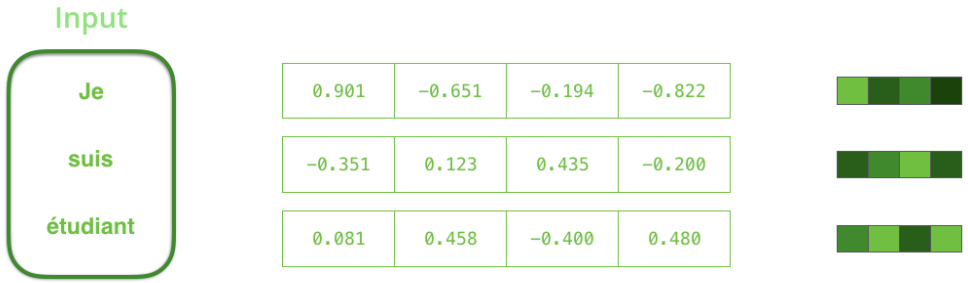
ENCODER – DECODER MACHINE TRANSLATION



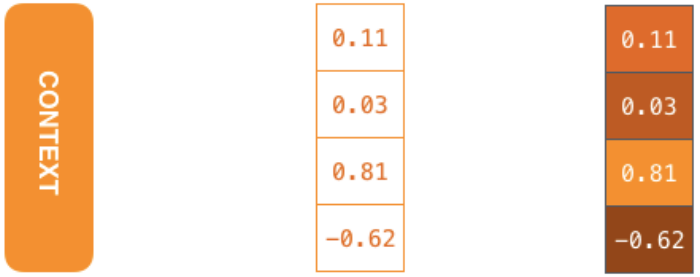
Credit: https://jalammar.github.io/

ENCODER – DECODER (SEQ2SEQ) MODEL

- **Input word:** Word embeddings



- **Context vector**
  - An array of real numbers with dimension the number of hidden units in the encoder (typical sizes are 256, 512 or 1024)



TRAINING A ENCODER-DECODER MODEL

training data typically consists of sets of sentences and their translations concatenated with a separator token.

Encoder-decoder architectures are trained end-to-end, just as with the RNN language models. The network is given the source text and then starting with the separator token is trained autoregressively to predict the next word

INFERENCE FROM ENCODER-DECODER MODEL

Inference:

- During inference decoder uses its own estimated output y_t as the input for the next time step x_{t+1}.

- Thus, the decoder will tend to deviate more and more from the gold target sentence as it keeps generating more tokens

ADVANTAGES OF NMT OVER SMT

- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized
- Requires much less human engineering effort
  - No feature engineering
  - Same method for all language pairs

DISADVANTAGES OF NMT COMPARED TO SMT

NMT directly calculates $P(y|x)$ while SMT breaks it down in to smaller parts

- NMT is less interpretable
  - Hard to debug
- NMT is difficult to control
  - For example, can't easily specify rules or guidelines for translation
  - Safety concerns!

LIMITATIONS OF THE ENCODER – DECODER ARCHITECTURE

- **Weakness**
  the influence of the context vector (c) will wane as the output sequence is generate.

- **Solution:**
  make the context vector available at each step in the decoding process by adding it as a parameter to the computation of the current hidden state

DECODER – ENCODER ARCHITECTURE

- The context vector turned out to be a bottleneck for these types of models.
- Its challenging for the models to deal with long sentences.

- **Solution:**
  - Attention
    - Bahdanau et al., 2014 introduced
    - Luong et al., 2015. refined
  - Attention allows the model to focus on the relevant parts of the input sequence as needed.
    - highly improved the quality of machine translation systems.

Attention

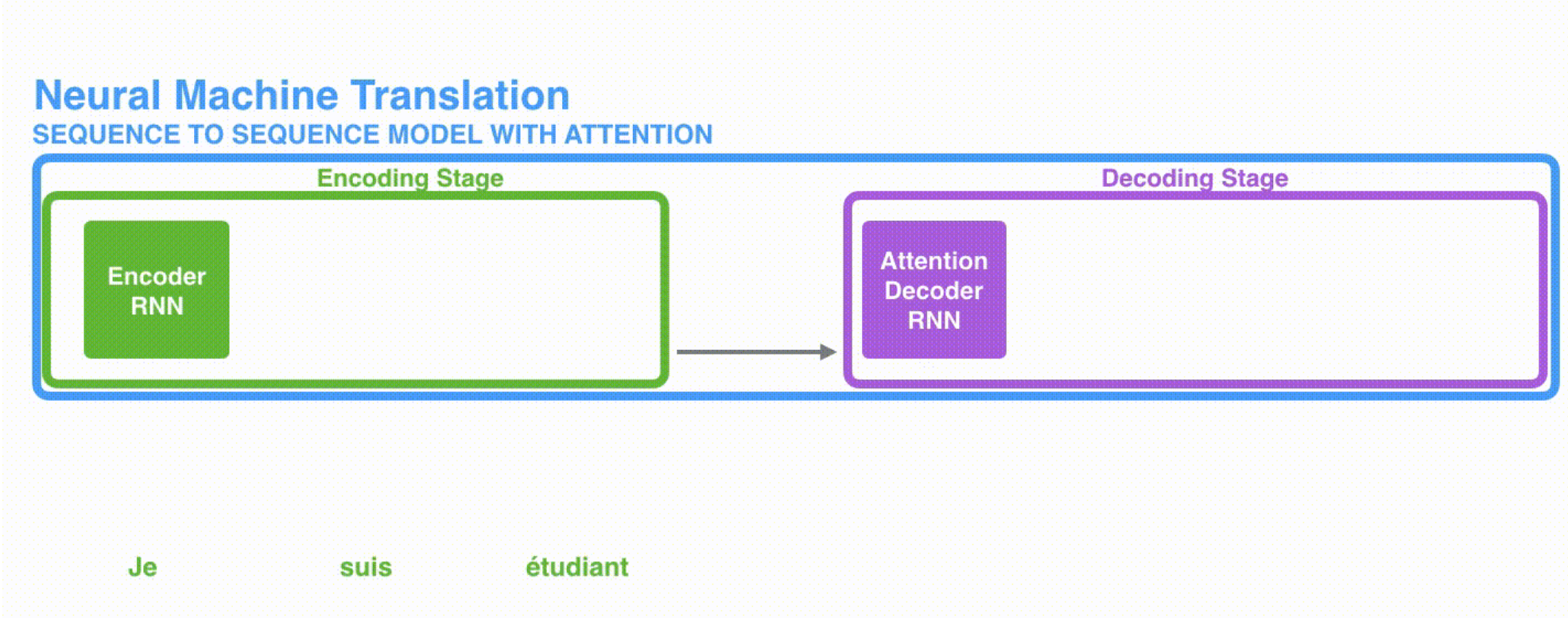
ATTENTION

"One important property of human perception is that one does not tend to process a whole scene in its entirety at once. Instead humans focus attention selectively on parts of the visual space to acquire information when and where it is needed, and combine information from different fixation over time to build up an internal representation of the scene, guiding future eye movements and decision making."

- Recurrent Models of visual Attention

HOW ATTENTION DIFFER FROM CLASSIC SEQ2SEQ MODEL

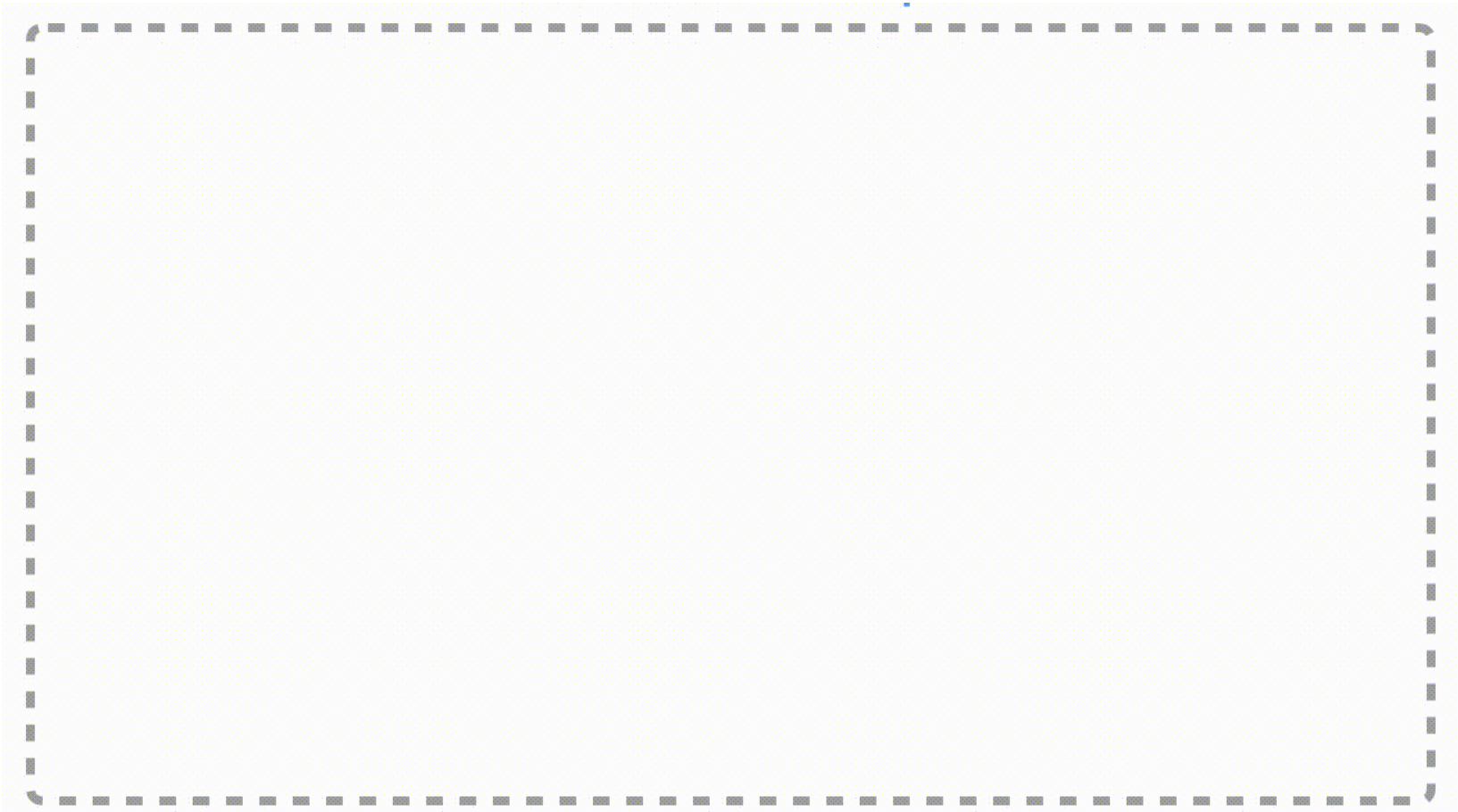
1. the encoder passes *all* the hidden states (context) to the decoder. Instead of passing the last hidden state of the encoding stage, the encoder passes *all* the hidden states to the decoder



HOW ATTENTION DIFFER FROM CLASSIC SEQ2SEQ MODEL

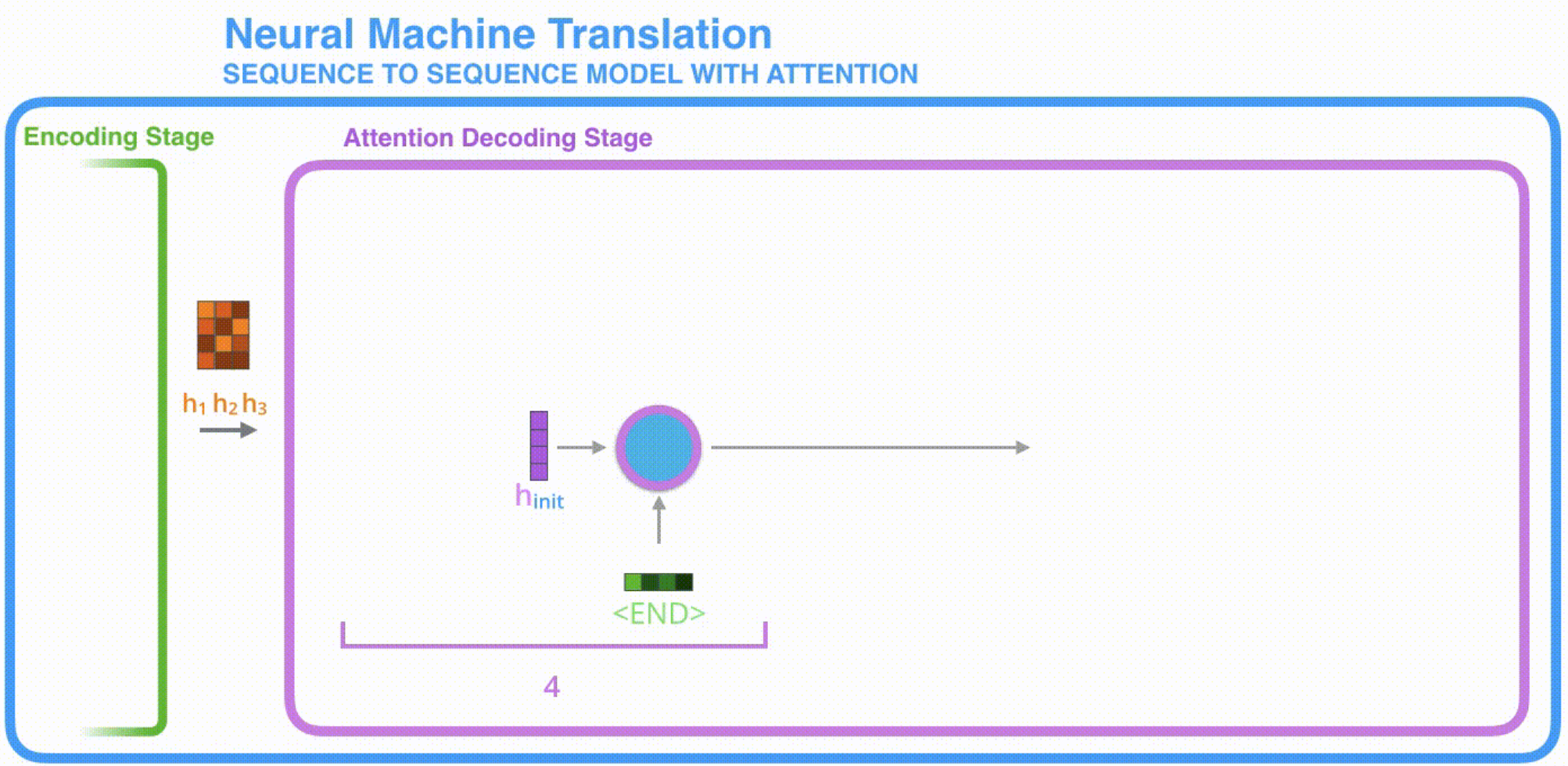
2. To focus on the relevant parts of the input decoder does the following
   i. Evaluate each encoder hidden states – each encoder hidden states is most associated with a certain word in the input sentence.
   ii. Assign a score to each hidden states (more later)
   iii. Multiply each hidden states by its softmaxed score, thus amplifying hidden states with high scores, and drowning out hidden states with low scores

Attention: Encoding process

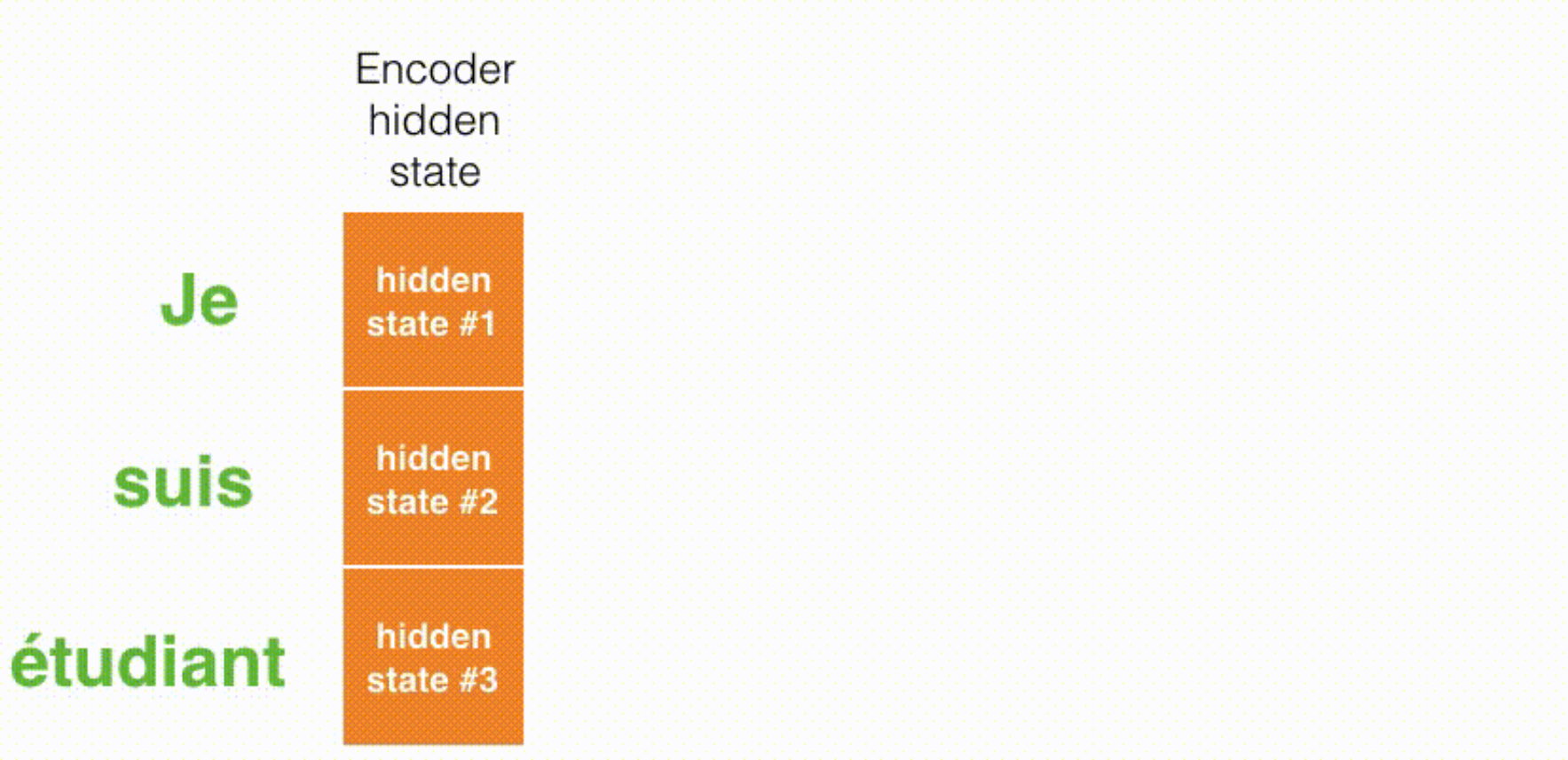


Attention

Scoring is done at each timestep on the decoder side



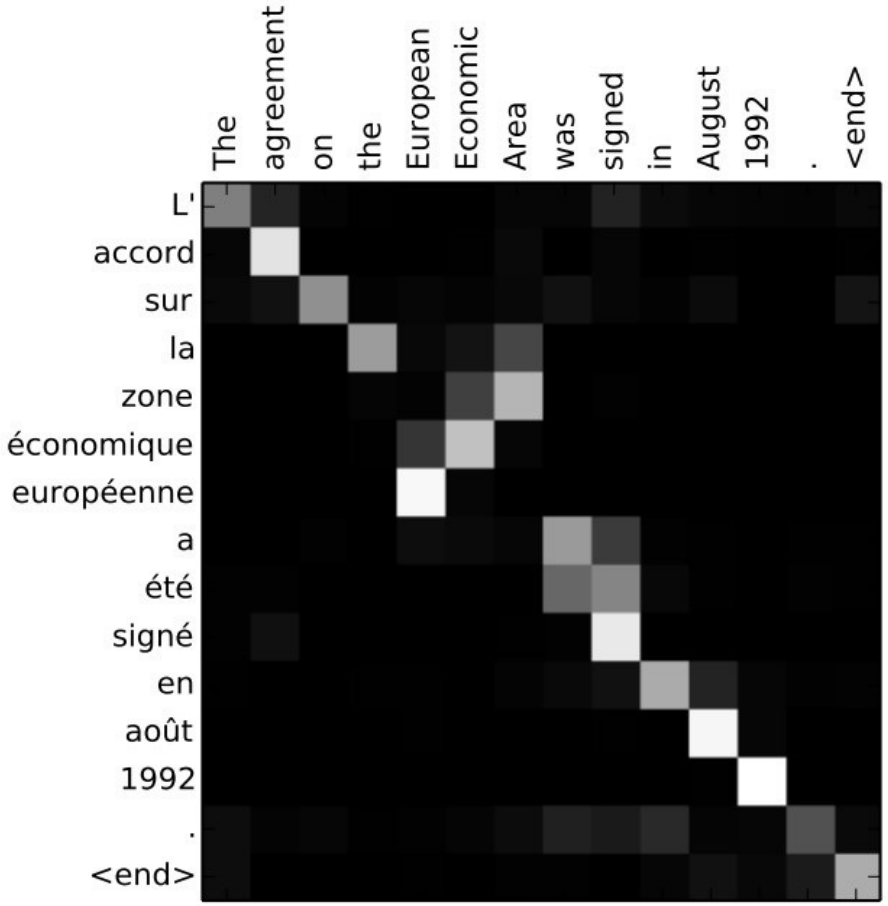
VISUALIZE ATTENTION



ATTENTION

Example of precision of the attention mechanism.

- The model learns how to align words in the language pair
- You can see how the model paid attention correctly
  - *European Economic Area*
  - *zone européenne économique*
  - Every other word in the sentence is in similar order.



REFERENCES