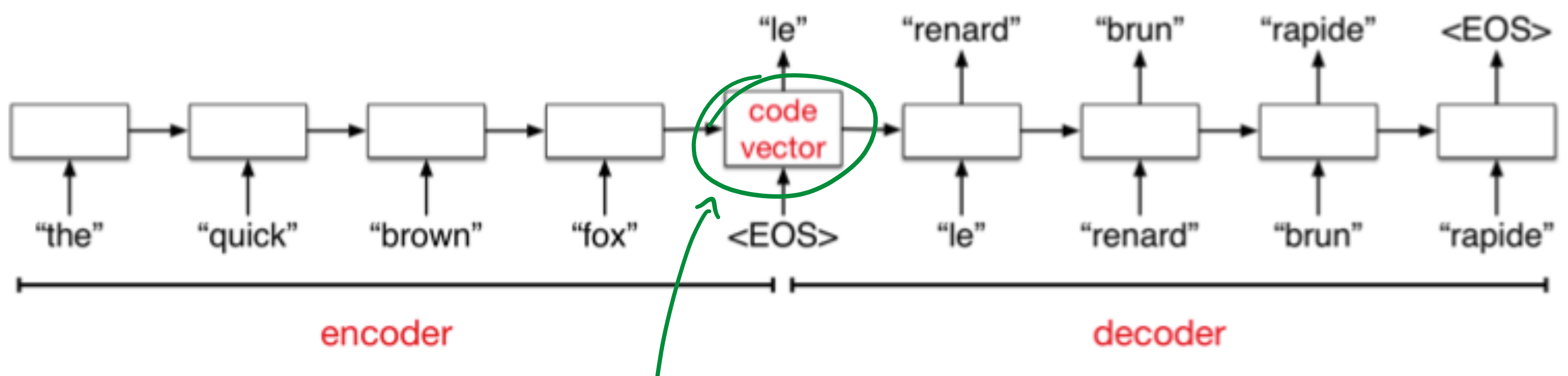
Video 32

Sunday, November 29, 2020 6:44 PM

Original encoder - decoder architecture for neural machine translation



memorize all relevant information about the input sentence

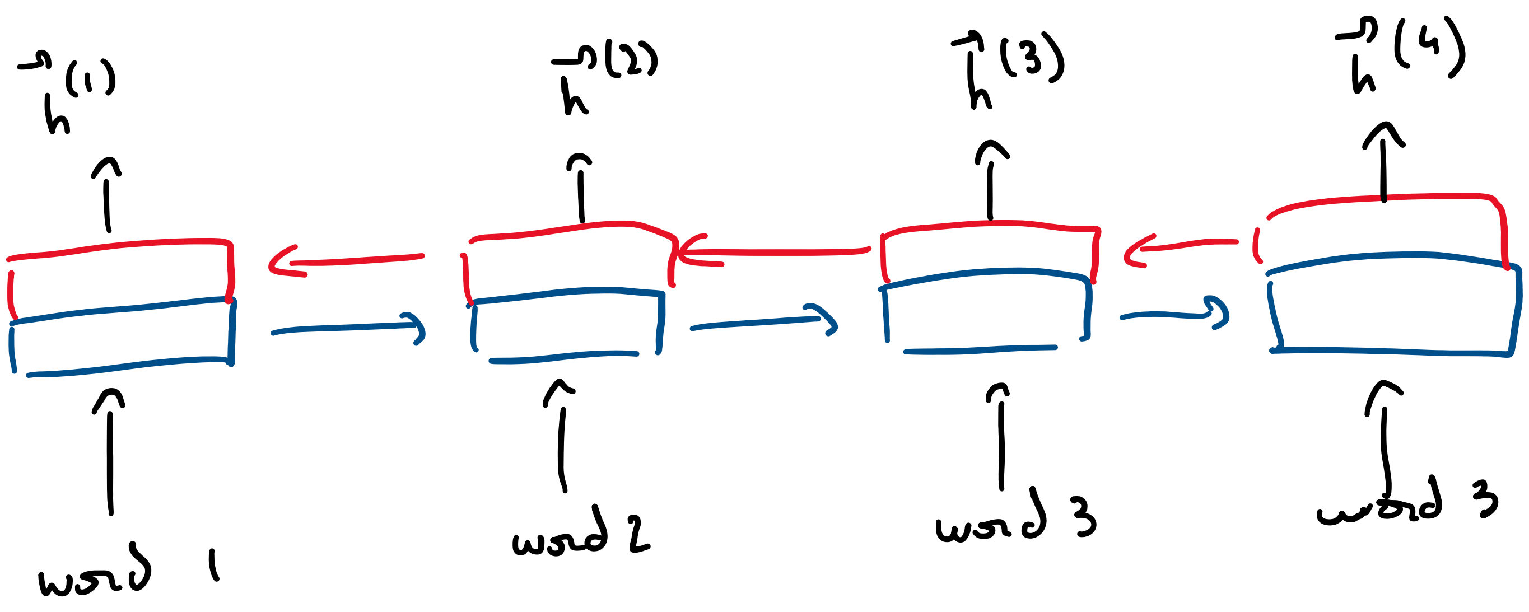
Attention - based modeling

→ look at input sequence (sentence) when generating text

Encoder : bidirectional RNN.

Goal : compute an annotation vector (which the decoder will consult when it generates the output sequence)

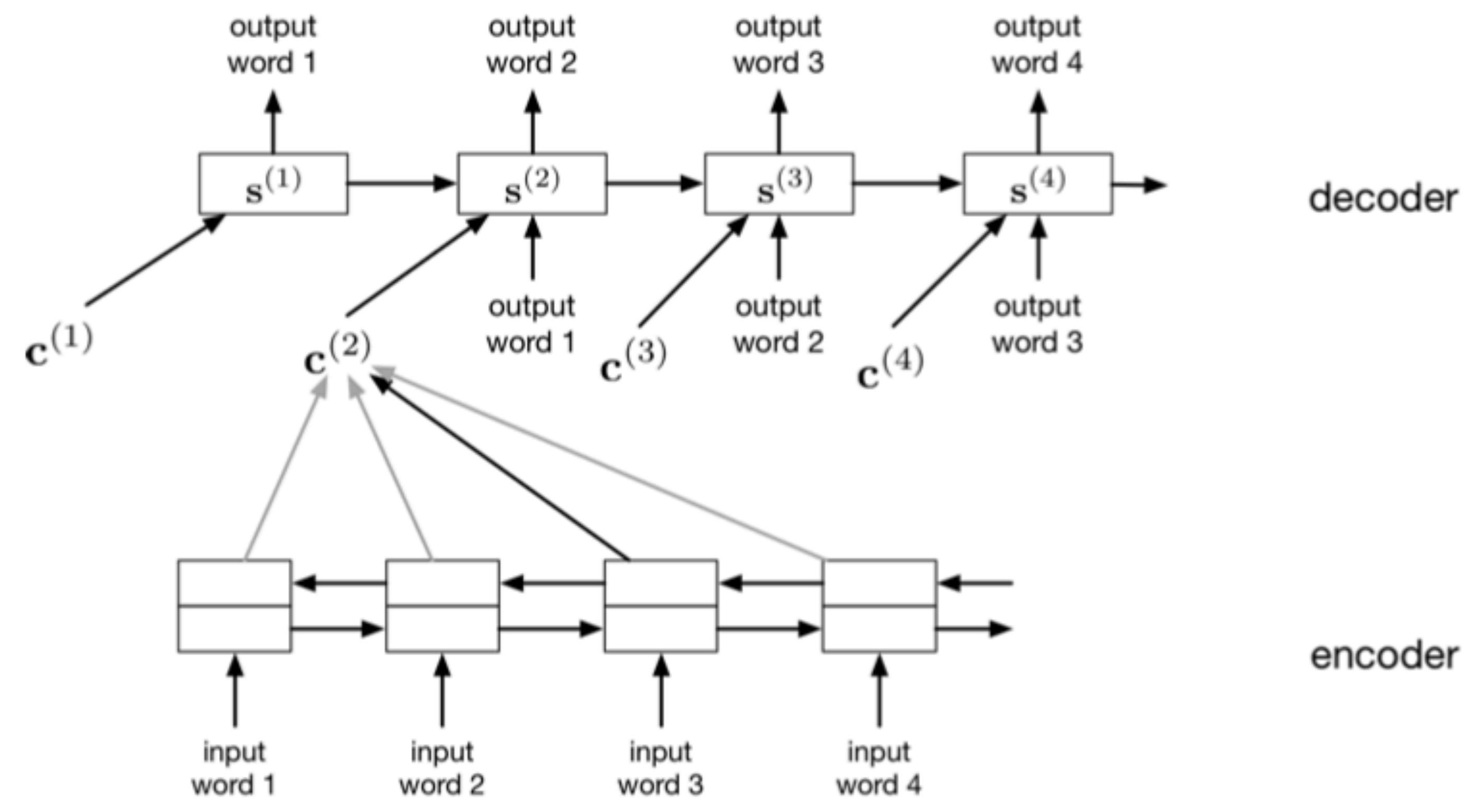
Bidirectional : two RNNs
↳ one processes the words in the forward order
↳ second ———————— backward order



Annotation vector : concatenation of the hidden units of the 2 RNNs.

Decoder : will get a context vector in addition to the words (generated up to current time step) as its inputs

context vector for time step 1



Soft - attention model :

$\vec{c}^{(i)}$: context vector for output time step i (decoder)

t : index of time steps in the input (encoder)

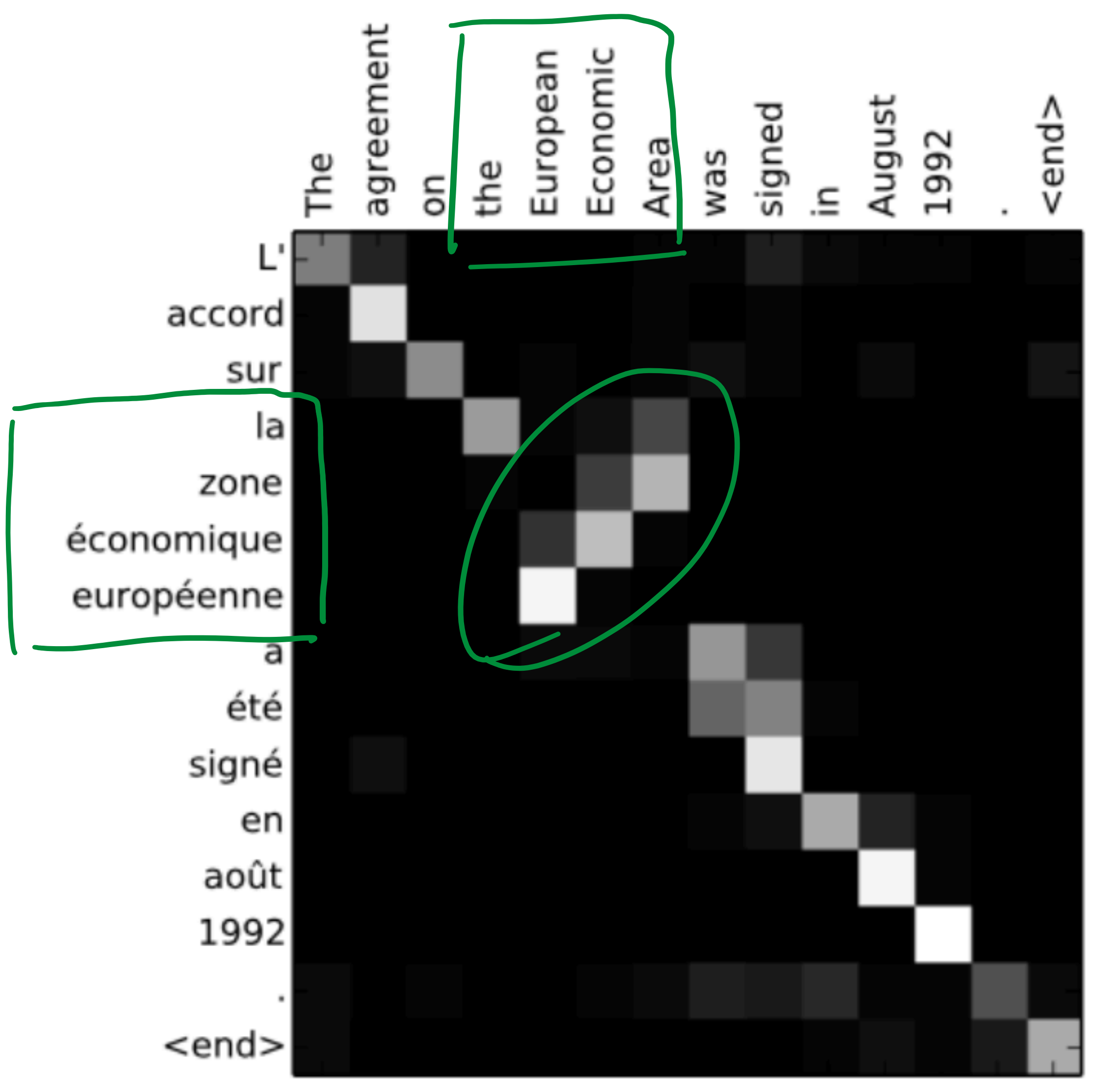
$$\vec{c}^{(i)} = \sum_t \underbrace{\alpha_{it}}_{\text{attention weight}} \underbrace{\vec{h}^{(t)}}_{\text{annotation vector}}$$

$$\alpha_{it} = \frac{e^{z_{it}}}{\sum_{t'} e^{z_{it'}}} \quad \text{where} \quad z_{it} = a\left(\vec{s}^{(i-1)}, \vec{h}^{(t)}\right)$$

feed forward neural network

previous decoder hidden state

annotation vector (used for content_based addressing)


The
agreement
on
the
European
Economic
Area
was
signed
in
August
<end>
L'
accord
sur
la
zone
économique
européenne
a
été
signé
en
août
<end>