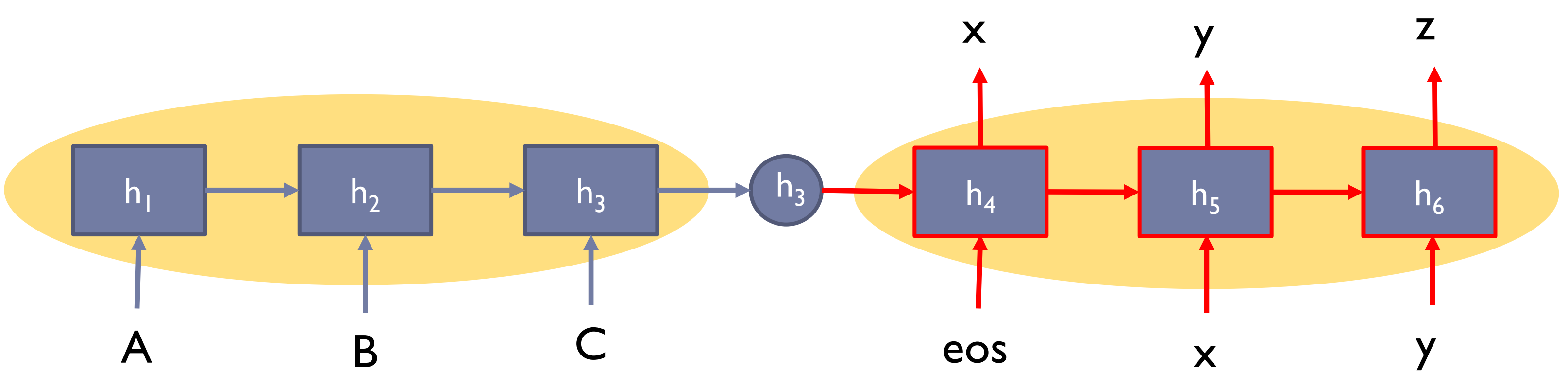
Attention Model

성균관대학교 소프트웨어학과
이 지 형

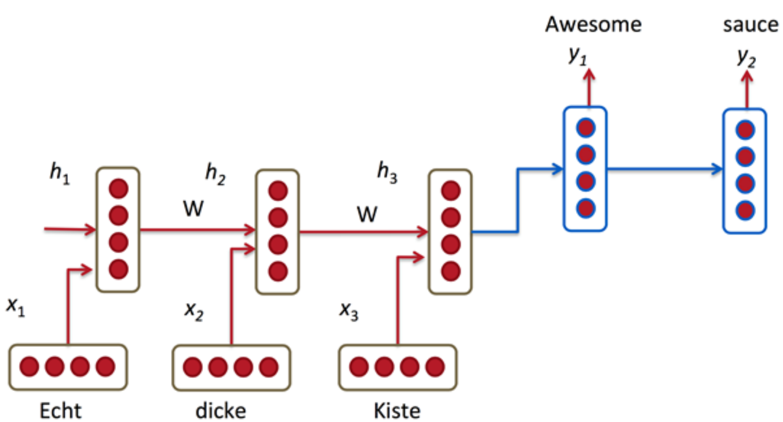
Sequence Generation

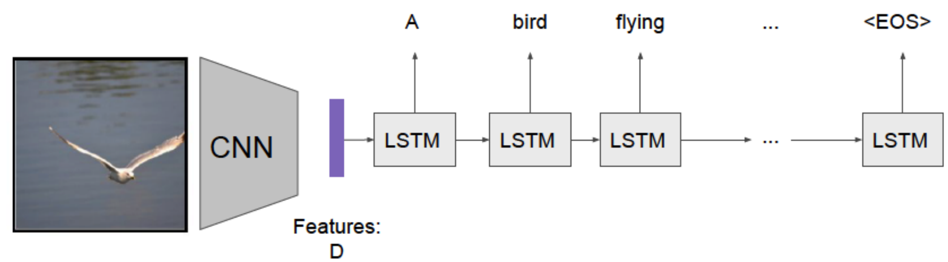
- **Encoder-Decoder Scheme**
  - **Encoder: compress input sequence into one vector**
    - h_3 is the vector representation of the given sequence
  - **Decoder: uses this vector to generate output**
    - It extracts necessary information only from the vector



Sequence Generation

- Encoder-Decoder Scheme
  - RNNs or CNNs can be used as Encoders
  - RNNs are usually used as Decoders



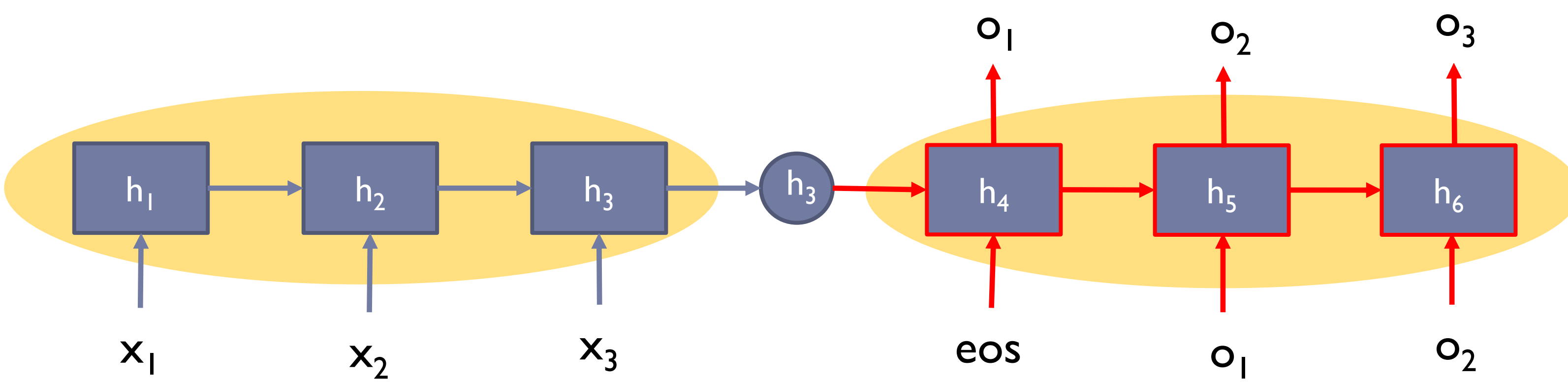


http://www.wildml.com/2016/01/attention-and-memory-in-deep-learning-and-nlp/

Attention Models, Amaia Salvador, 2017

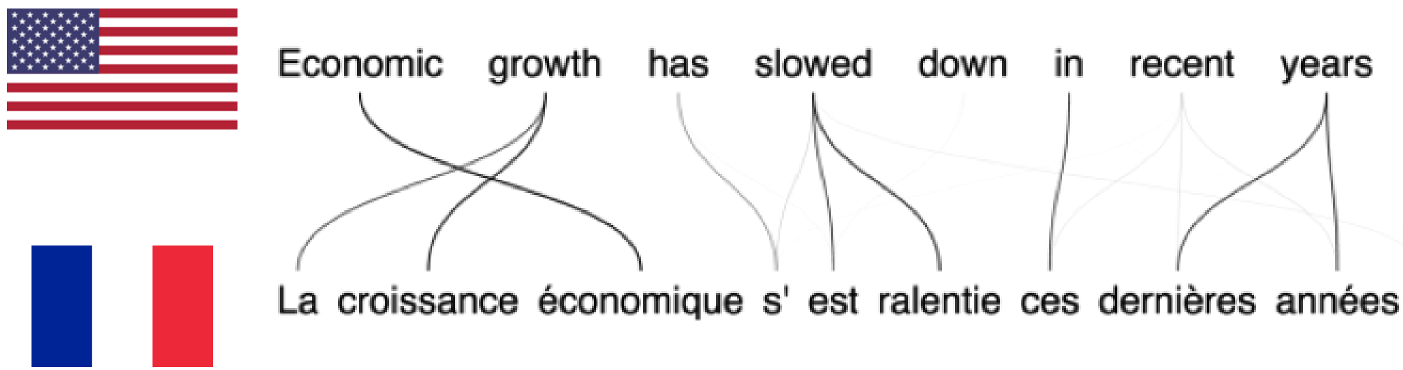
Sequence Generation

- **Challenges**
  - Hard for encoder to compress the whole source sentence into a single vector
  - Performance is degraded as the length of sentence increases
  - A single vector may not enough for decoder to generate correct words



Attention Model

- **Observation**
  - At every step, all the inputs are not equally useful

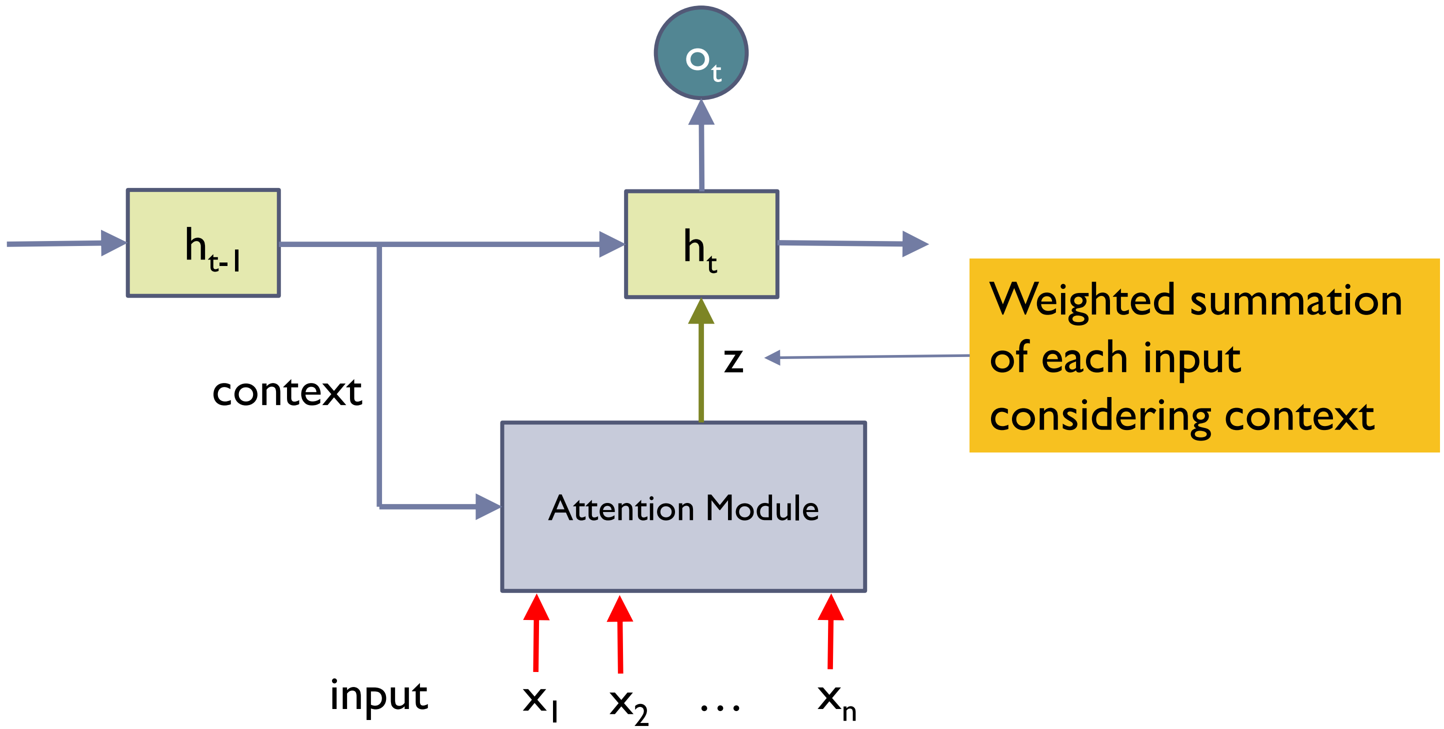


  - Inputs relevant to the context may be more useful

Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)

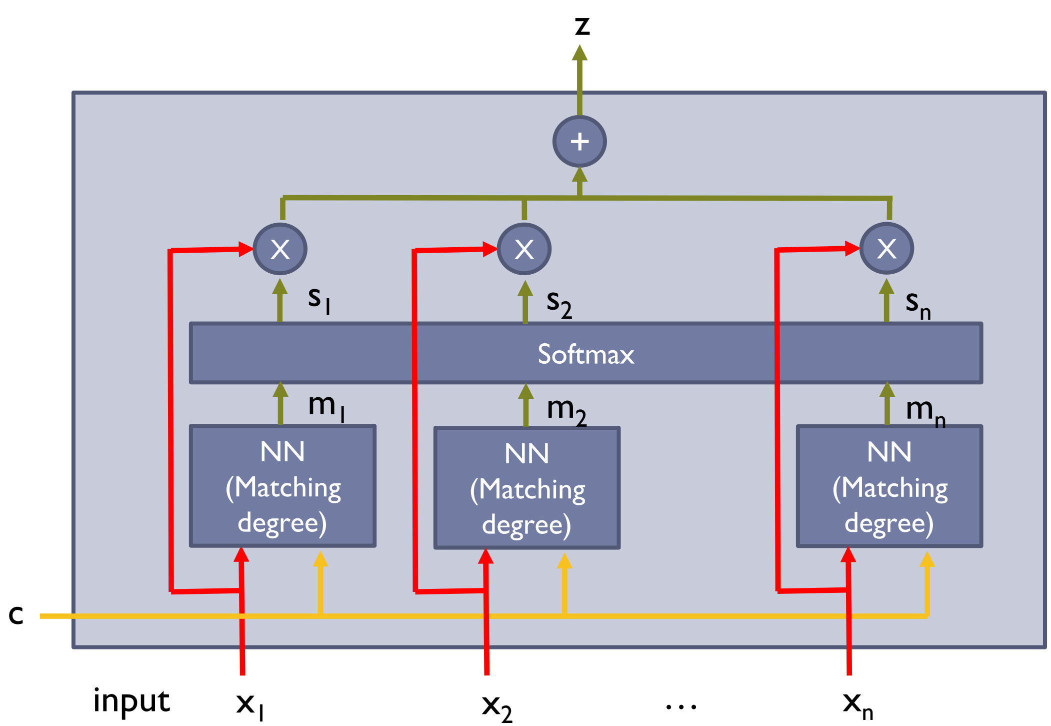
Attention Model

Overview



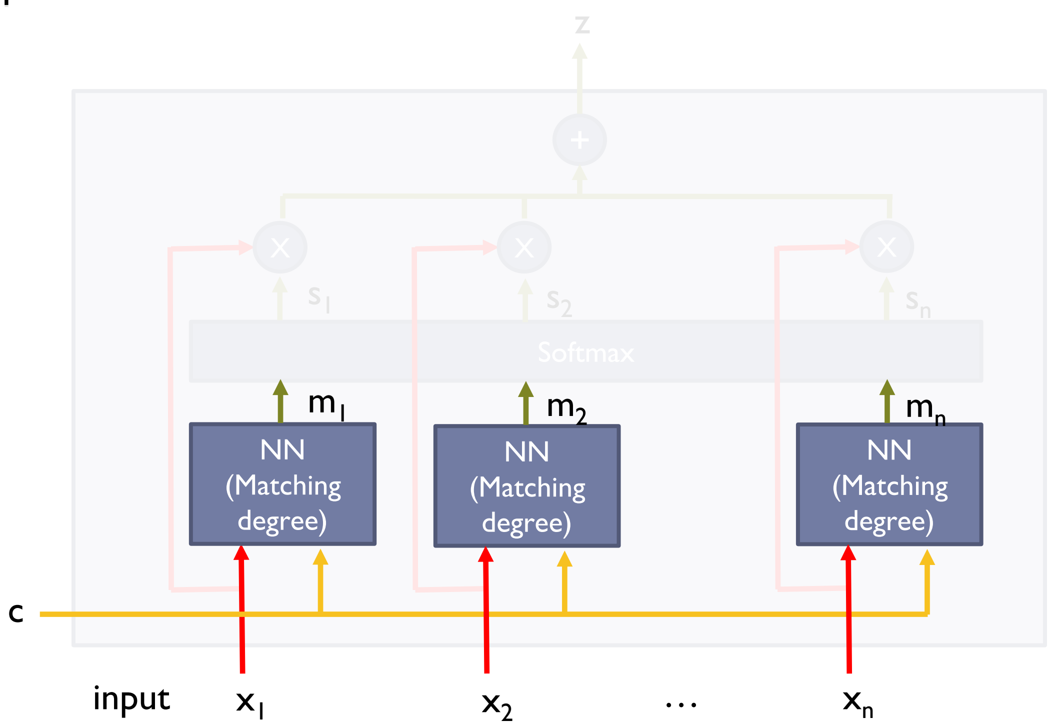
Attention Model

- **Attention Module**
  - All inputs share the same NN for matching degree



Attention Model

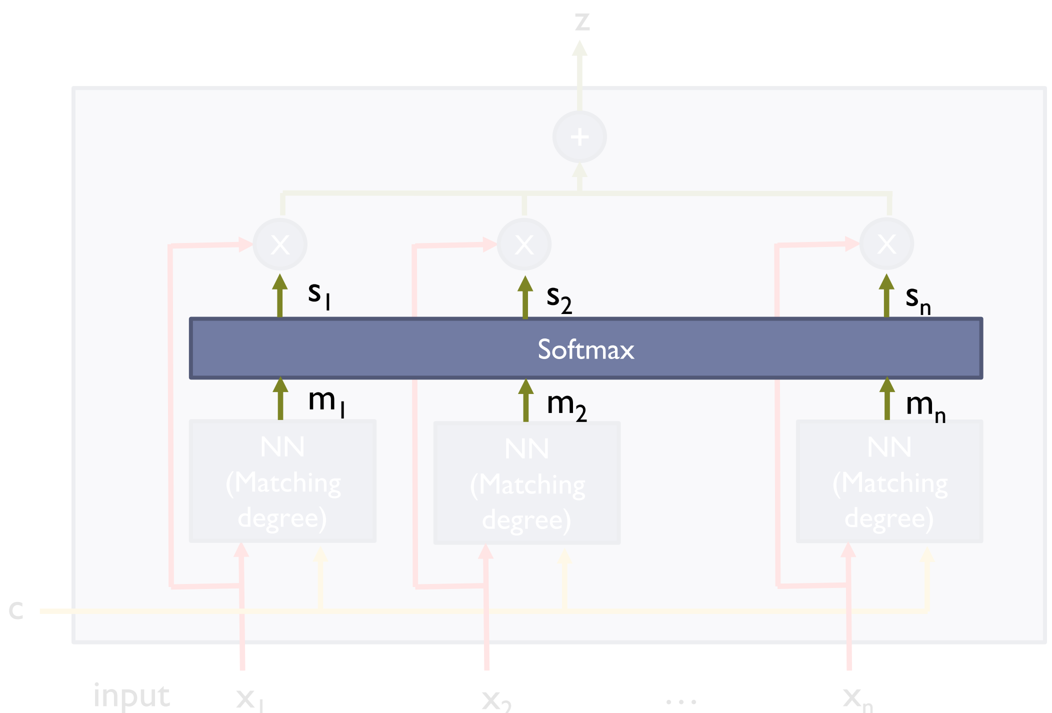
- **Step 1: Evaluating Matching Degree**
  - **Evaluating matching degree of each input to the context**
    - Produce scalar matching degree (Higher value is higher attention)
    - All inputs share the same NN



Attention Model

- ## Step 2: Normalizing Matching Degree

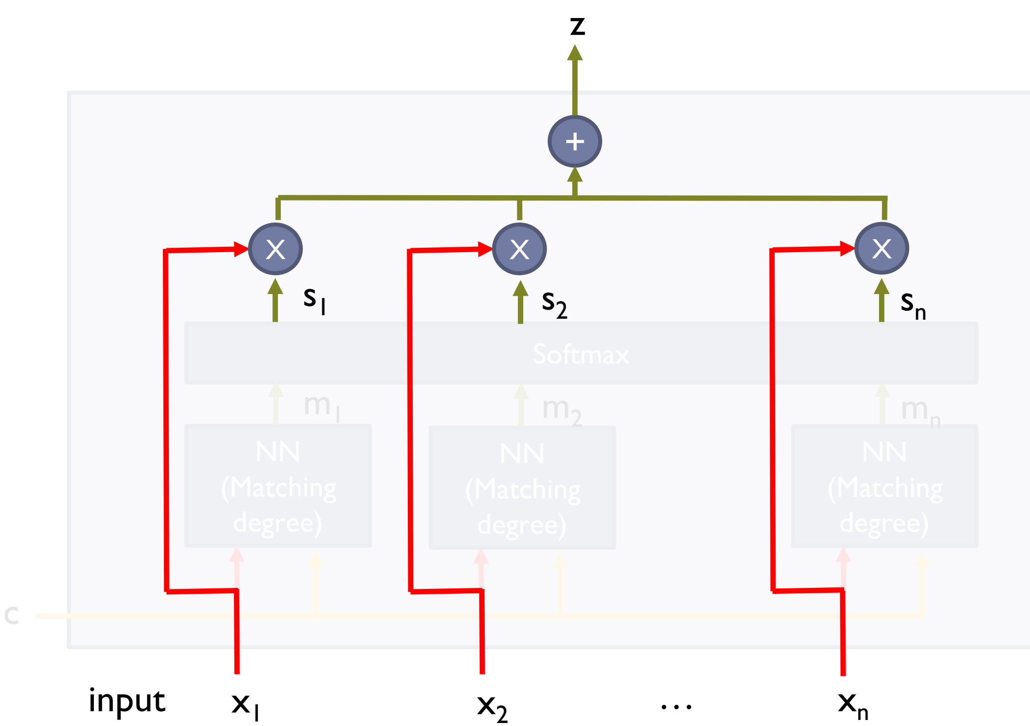
$$s_i = \frac{\exp(m_i)}{\sum_j exp(m_j)}$$



성균관대학교

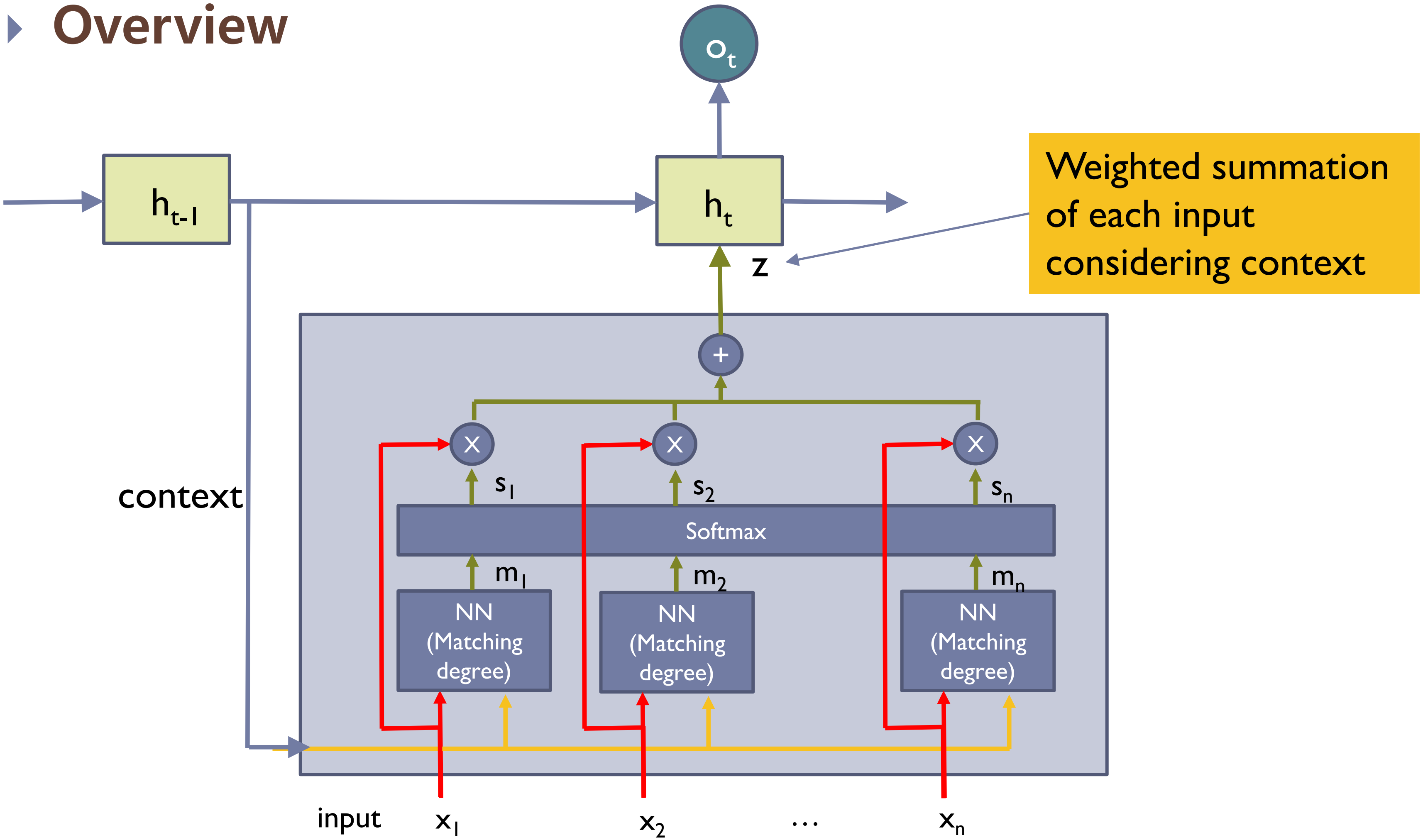
Attention Model

- **Step 3: Aggregating Inputs**
  - Each input is scaled by s_i and summed up into z
  - z is the input focused on the current context



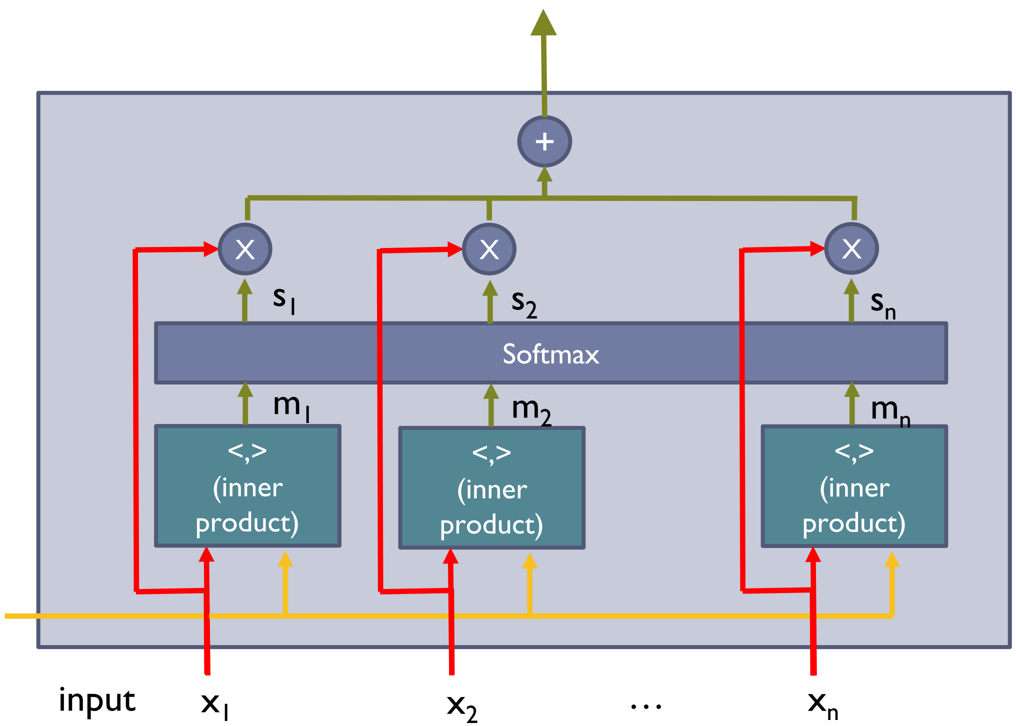
Attention Model

- **Overview**



Attention Model

- ## Variation
  - Matching NN can be replaced with the inner products of inputs and context



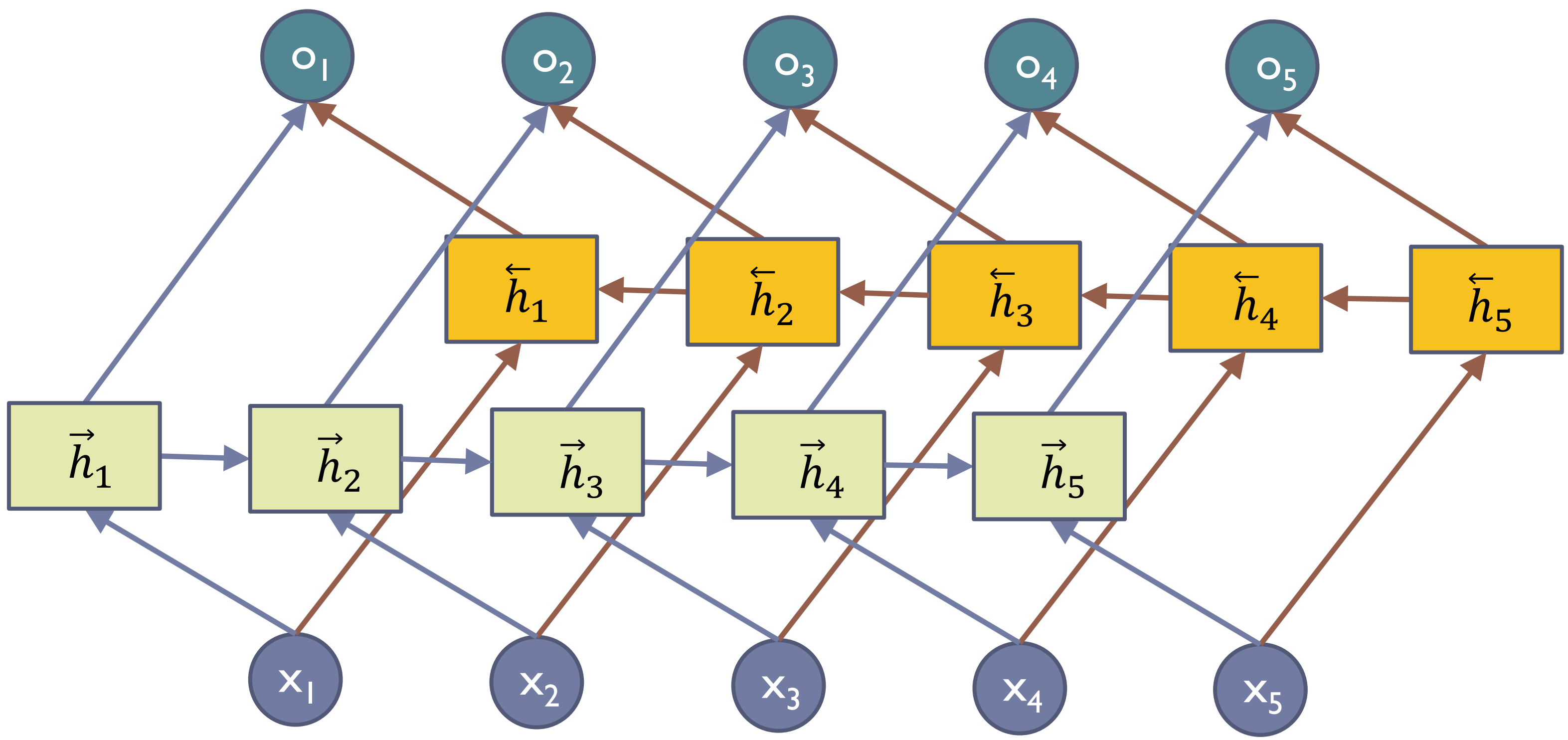
Attention Model

- **Bidirectional LSTM**

$h_i = [\overrightarrow{h}_i, \overleftarrow{h}_i]$ represents the past and future information

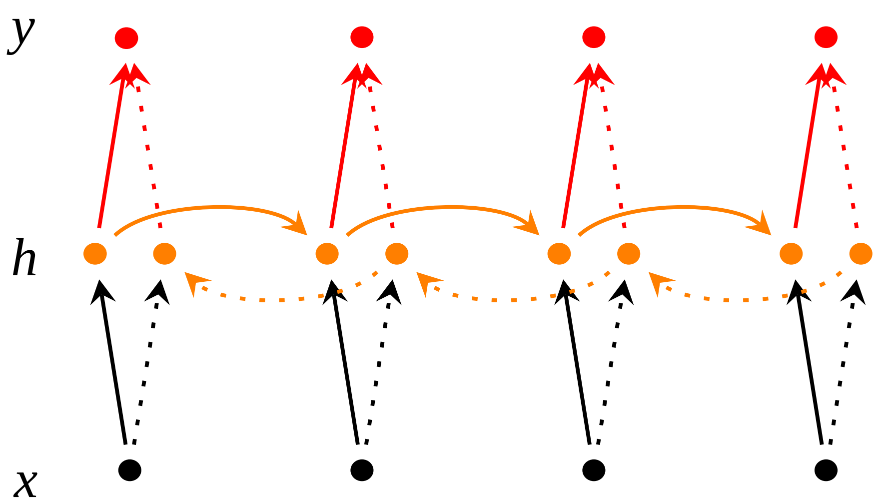
$\overrightarrow{h}_i$ represents the past information

$\overleftarrow{h}_i$ represents the future information



Attention Model

- ## Bidirectional LSTM



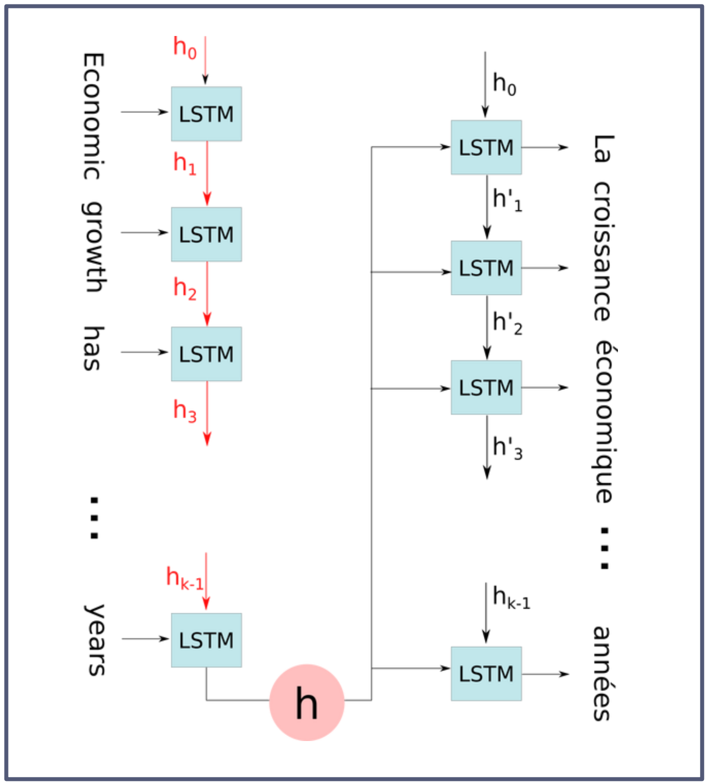
$$\overrightarrow{h}_t = f(\overrightarrow{W}x_t + \overrightarrow{V}\overrightarrow{h}_{t-1} + \overrightarrow{b})$$

$$\overleftarrow{h}_t = f(\overleftarrow{W}x_t + \overleftarrow{V}\overleftarrow{h}_{t+1} + \overleftarrow{b})$$

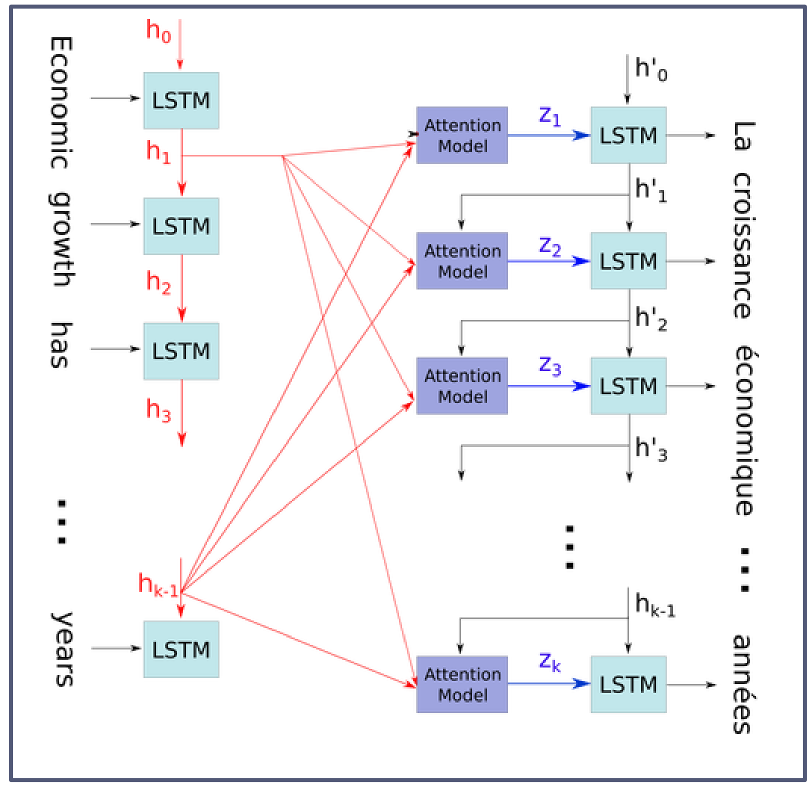
$$y_t = g(U[\overrightarrow{h}_t;\overleftarrow{h}_t] + c)$$

Attention Model

- **Example**



Encoder-decoder model

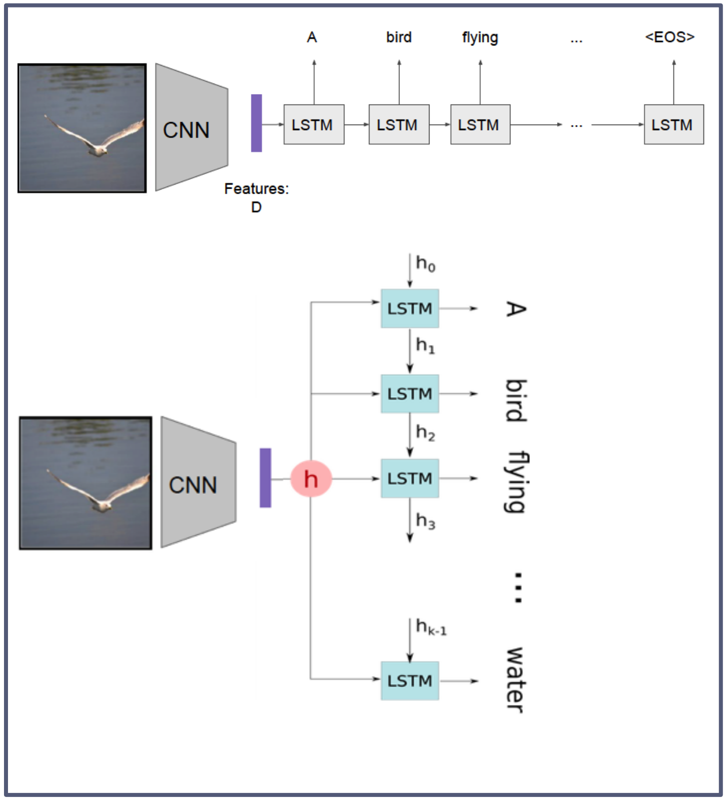


Attention based model

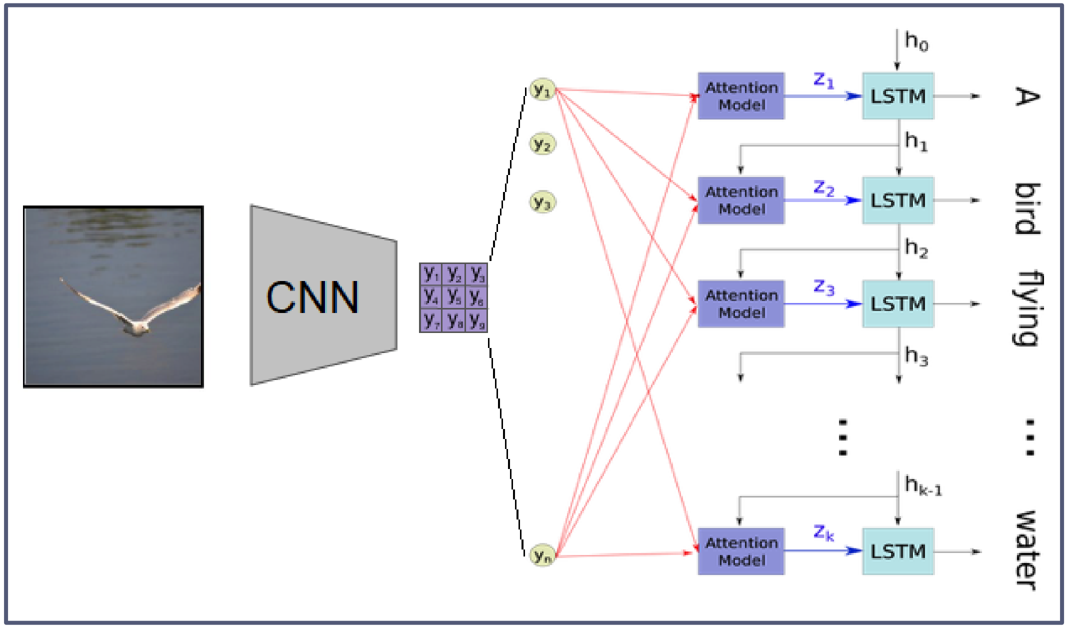
https://blog.heuritech.com/2016/01/20/attention-mechanism/

Attention Model

Example



Encoder-decoder model

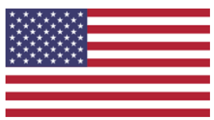


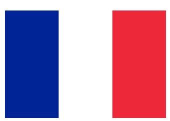
Attention based model

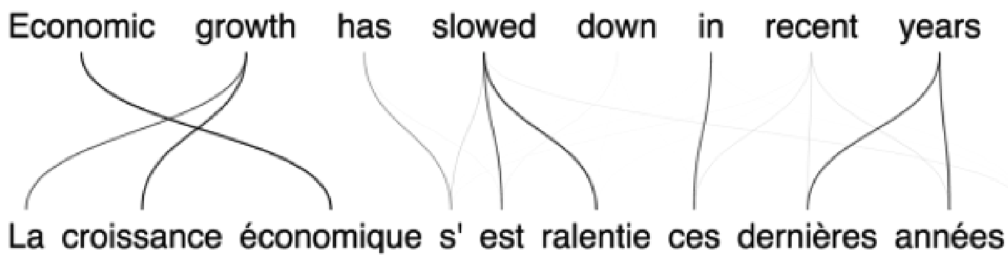
Attention Models, Amaia Salvador, 2017 https://blog.heuritech.com/2016/01/20/attention-mechanism/

Attention Model

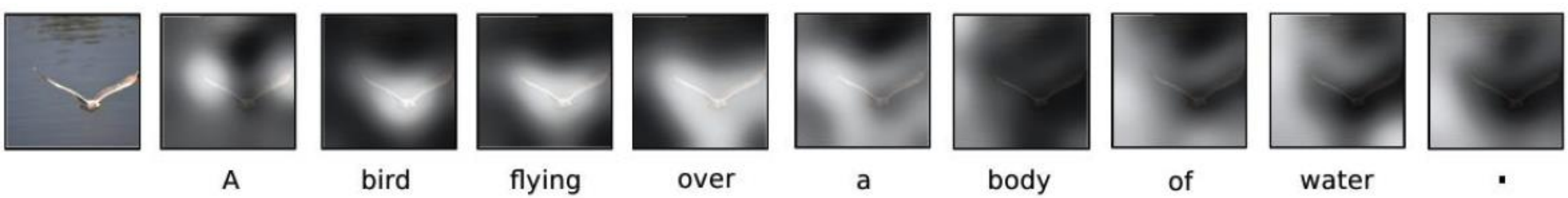
- One more advantage
  - We can interpret and visualize what the model is doing







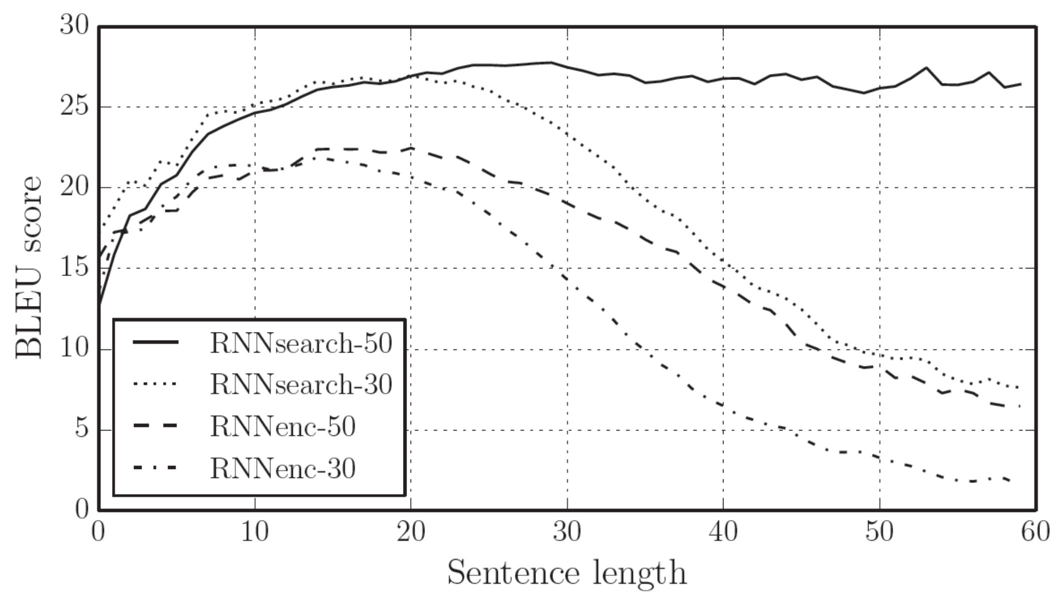
Kyunghyun Cho, "Introduction to Neural Machine Translation with GPUs" (2015)



Xu et al. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. ICML 2015

Attention is Great!

- **RNNsearch-50 is a neural machine translation model with the attention mechanism trained on all the sentence pairs of length at most 50.**
  - Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," ICLR 2015



Attention is Great!

- **Attention significantly improves NMT performance.**
  - It’s very useful to allow decoder to focus on certain parts of the source.
- **Attention solves the bottleneck problem.**
  - Attention allows decoder to look directly at source; bypass bottleneck.
- **Attention helps with vanishing gradient problem.**
  - Provides shortcut to faraway states.
- **Attention provides some interpretability.**
  - By inspecting attention distribution, we can see what the decoder was focusing on.
  - We get alignment for free!
  - This is cool because we never explicitly trained an alignment system
  - The network just learned alignment by itself.

Question and Answer