29-10–2024

Self-Attention network: transformer

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

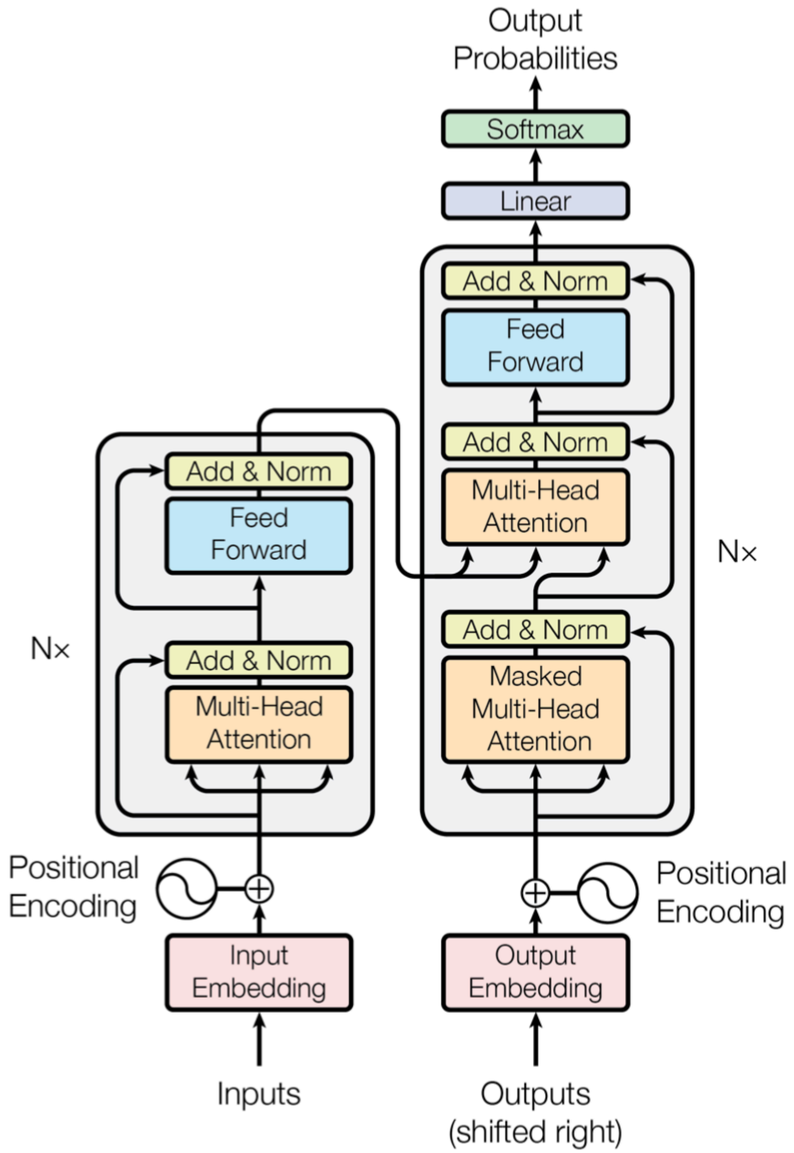
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.



Multi-head self-attention

- Different aspects of relationships
  - Syntactic
  - Semantic
  - Discourse
  - …

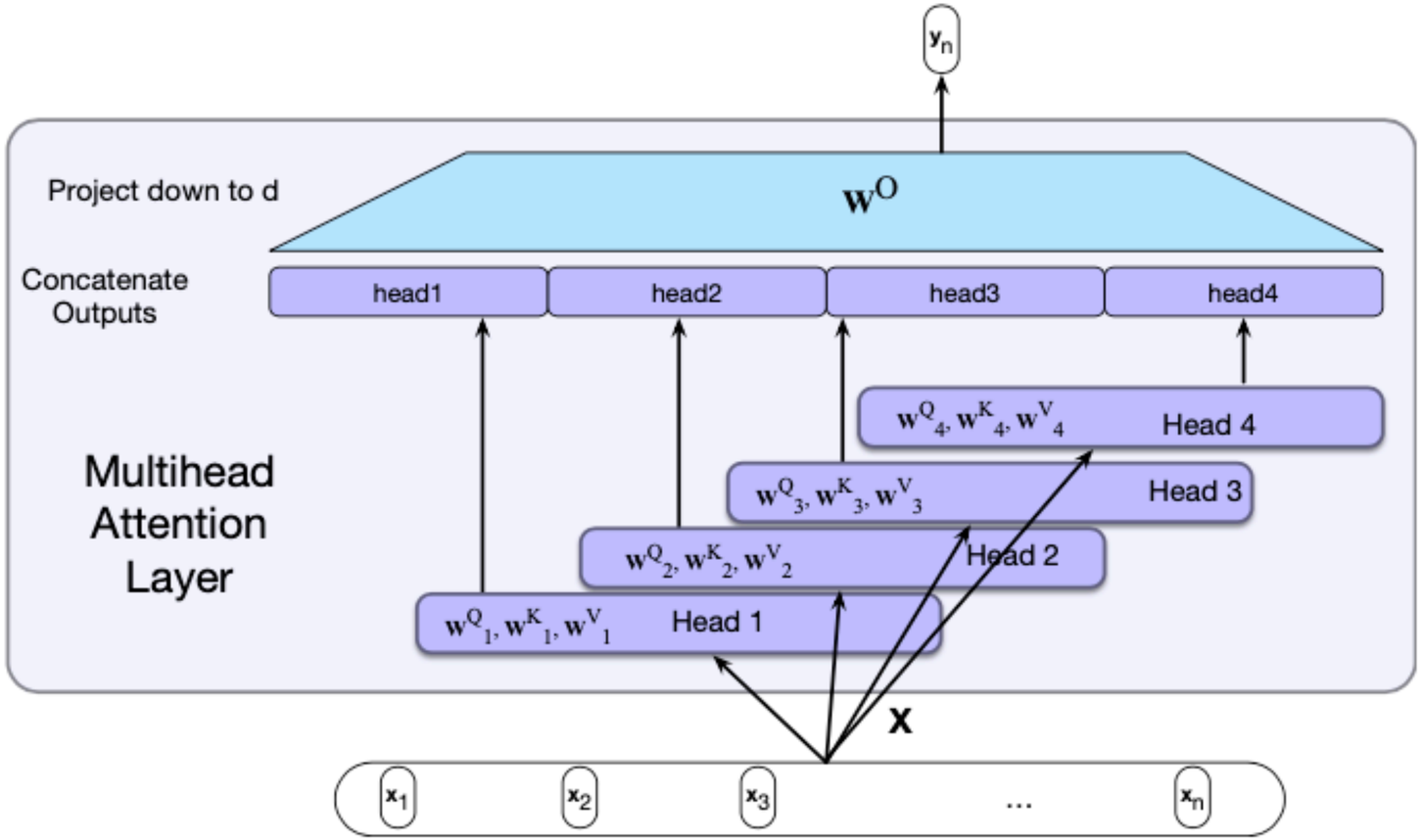


Image: Jurafsky & Martin “Speech and Language Processing, 3rd ed., 2023

Positional embedding

- Sine and cosine function of different frequencies:
  - $PE_{(pos,2i)} = sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right)$
  - $PE_{(pos,2i+1)} = cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right)$

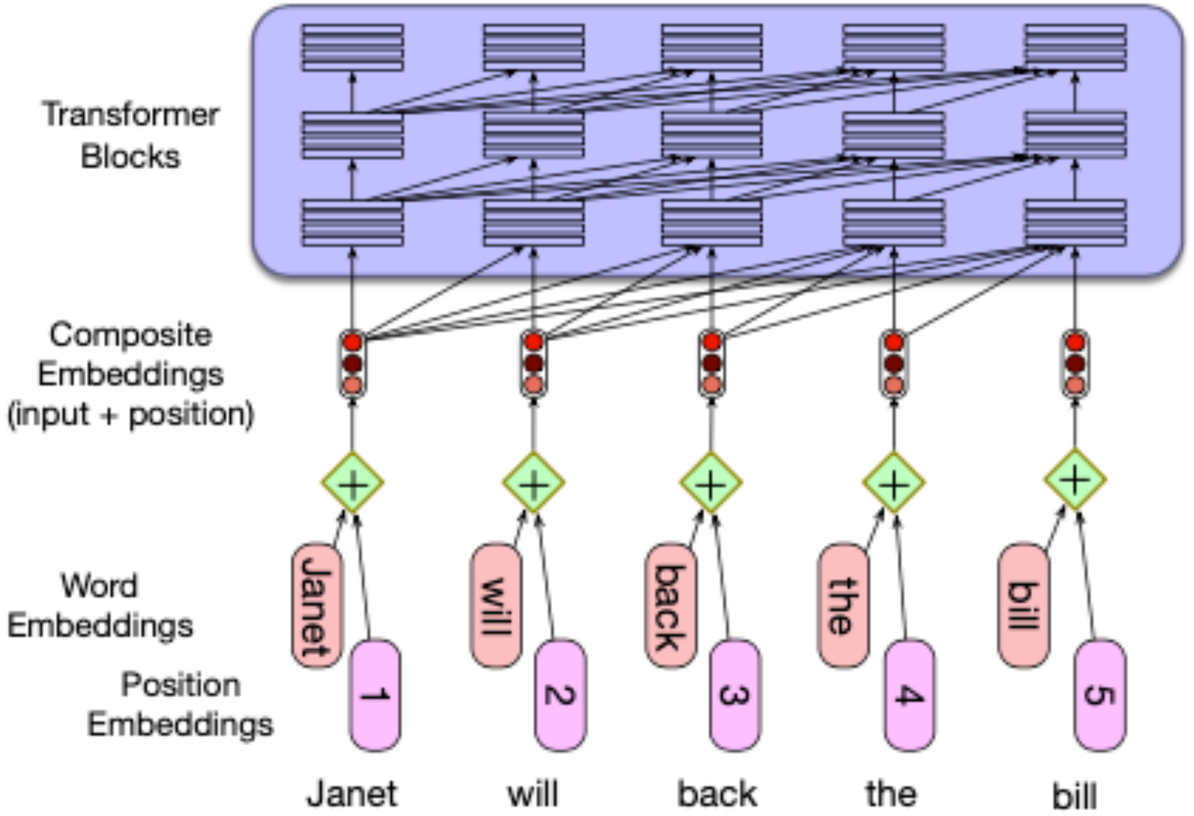


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Transformer as language model

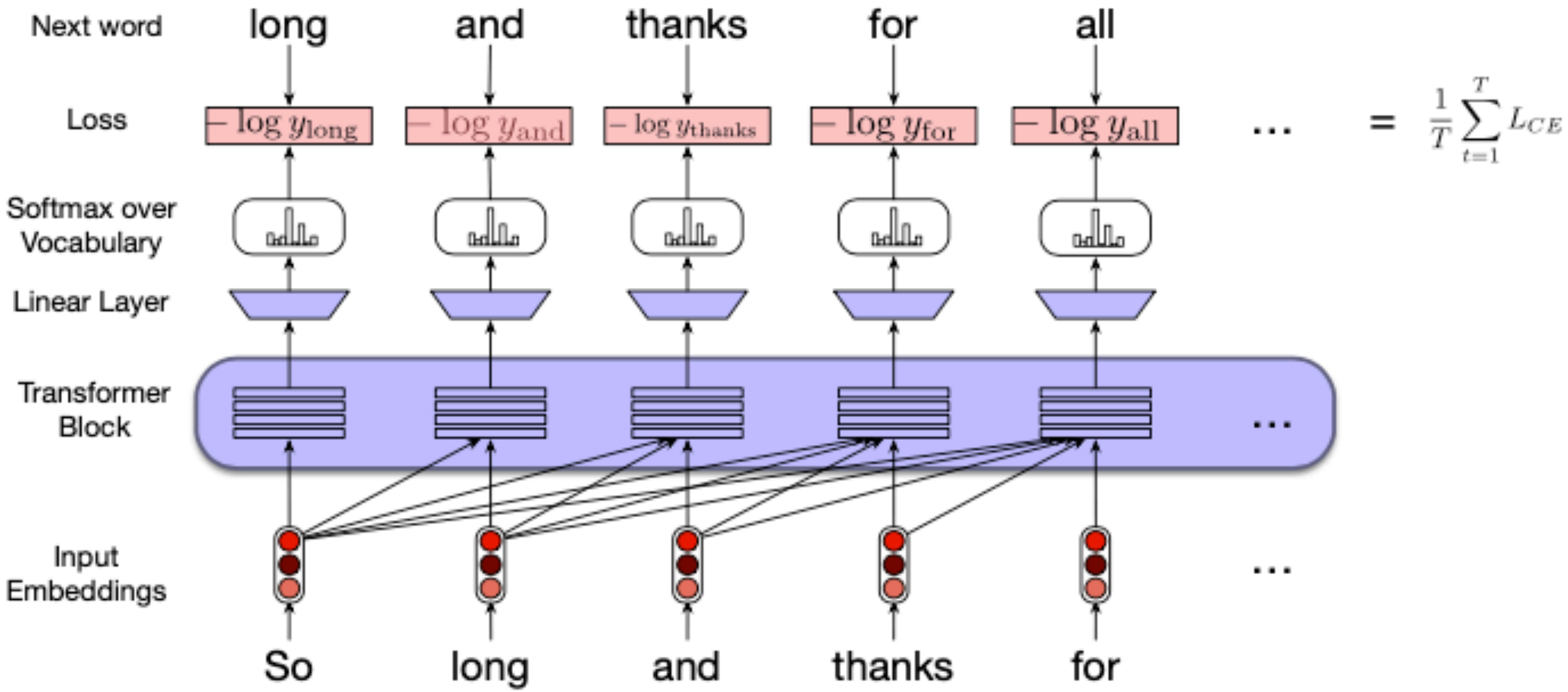


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Summarization

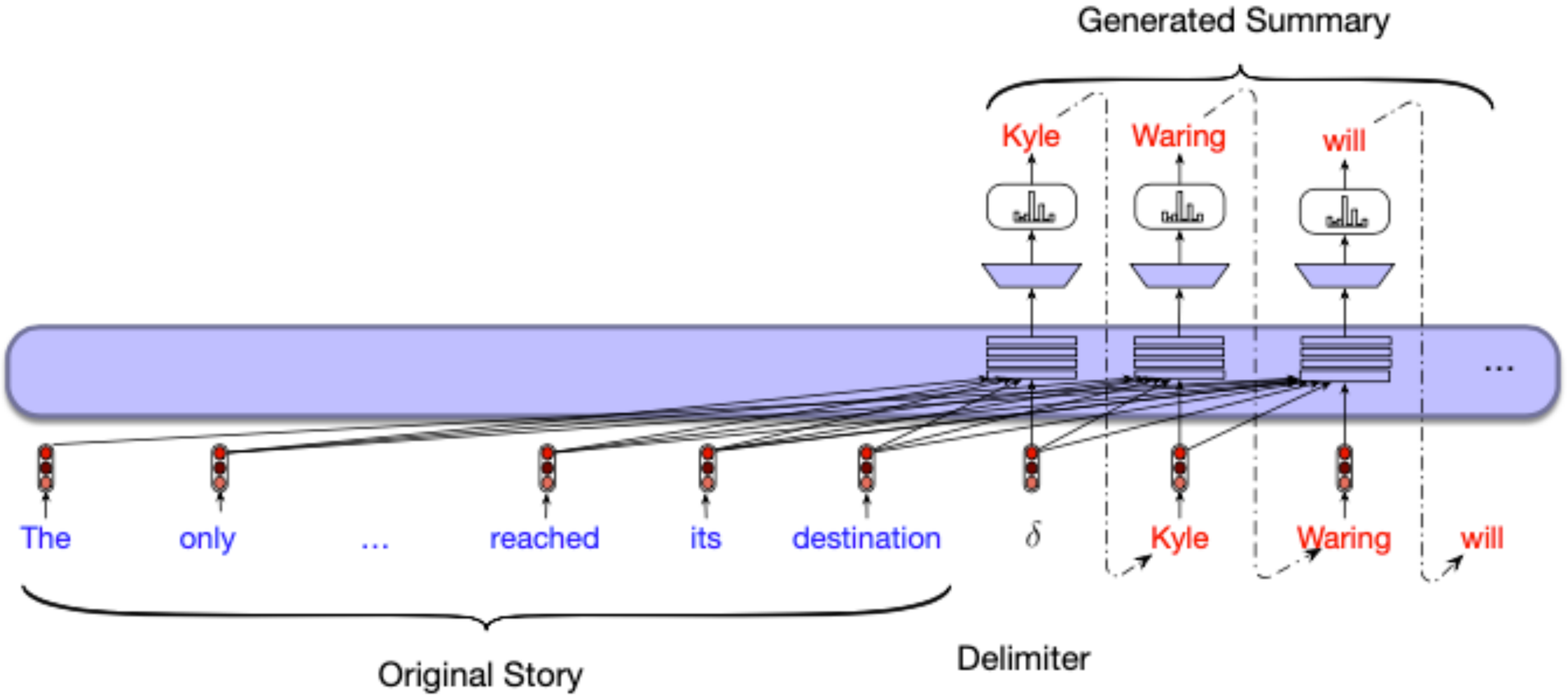


Image: Jurafsky & Martin “Speech and Language Processing, 3rd ed., 2023

Text generation

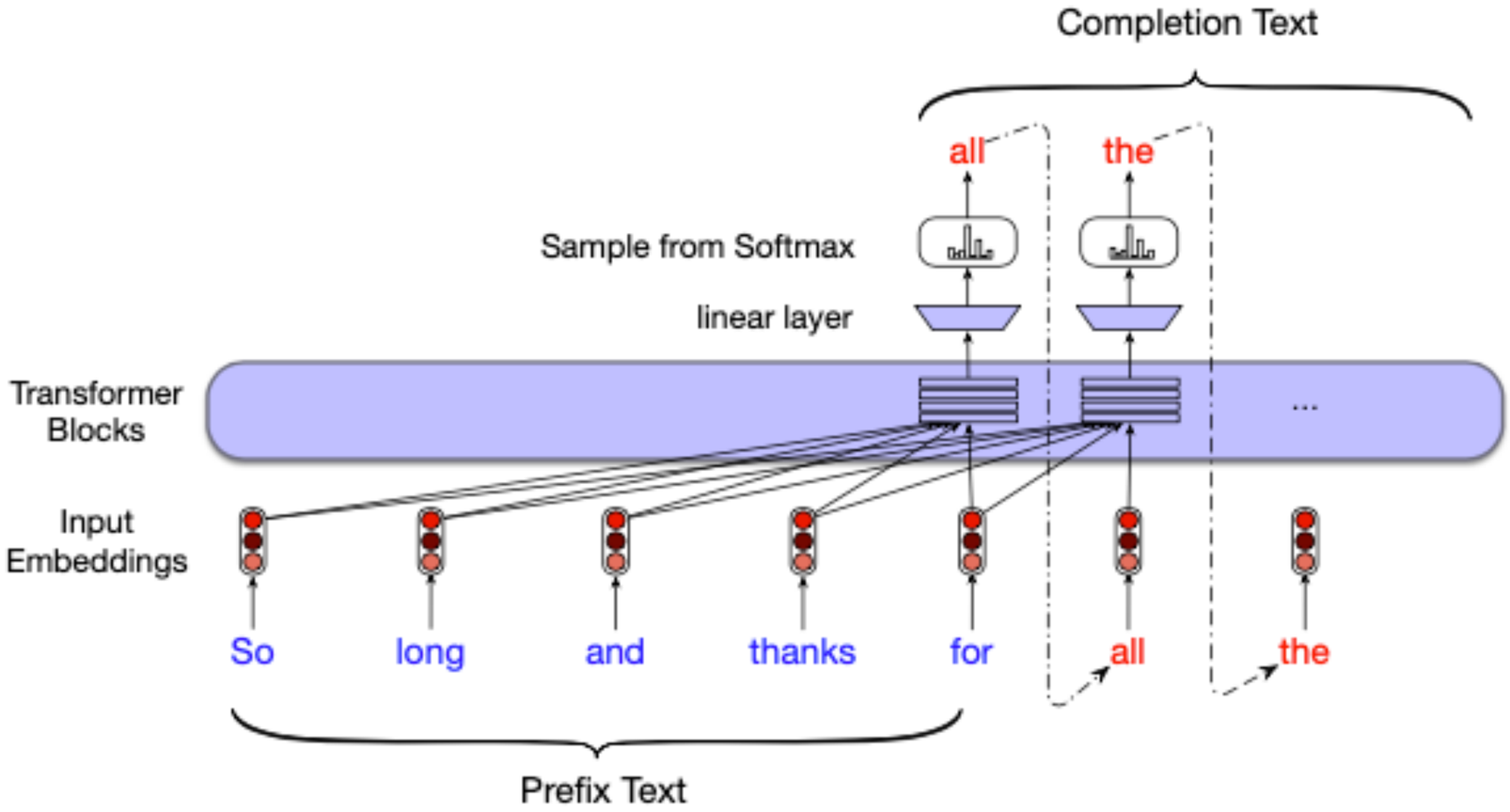


Image: Jurafsky & Martin “Speech and Language Processing, 3rd ed., 2023

Machine Translation and Encoder-Decoder Models

- Encoder: takes an input sequence and creates a contextualized representation of it
  - Learn context of the input sequence
    - RNN, LSTM, Self-attention
- Decoder: Generate a task specific output sequence
  - Language translation

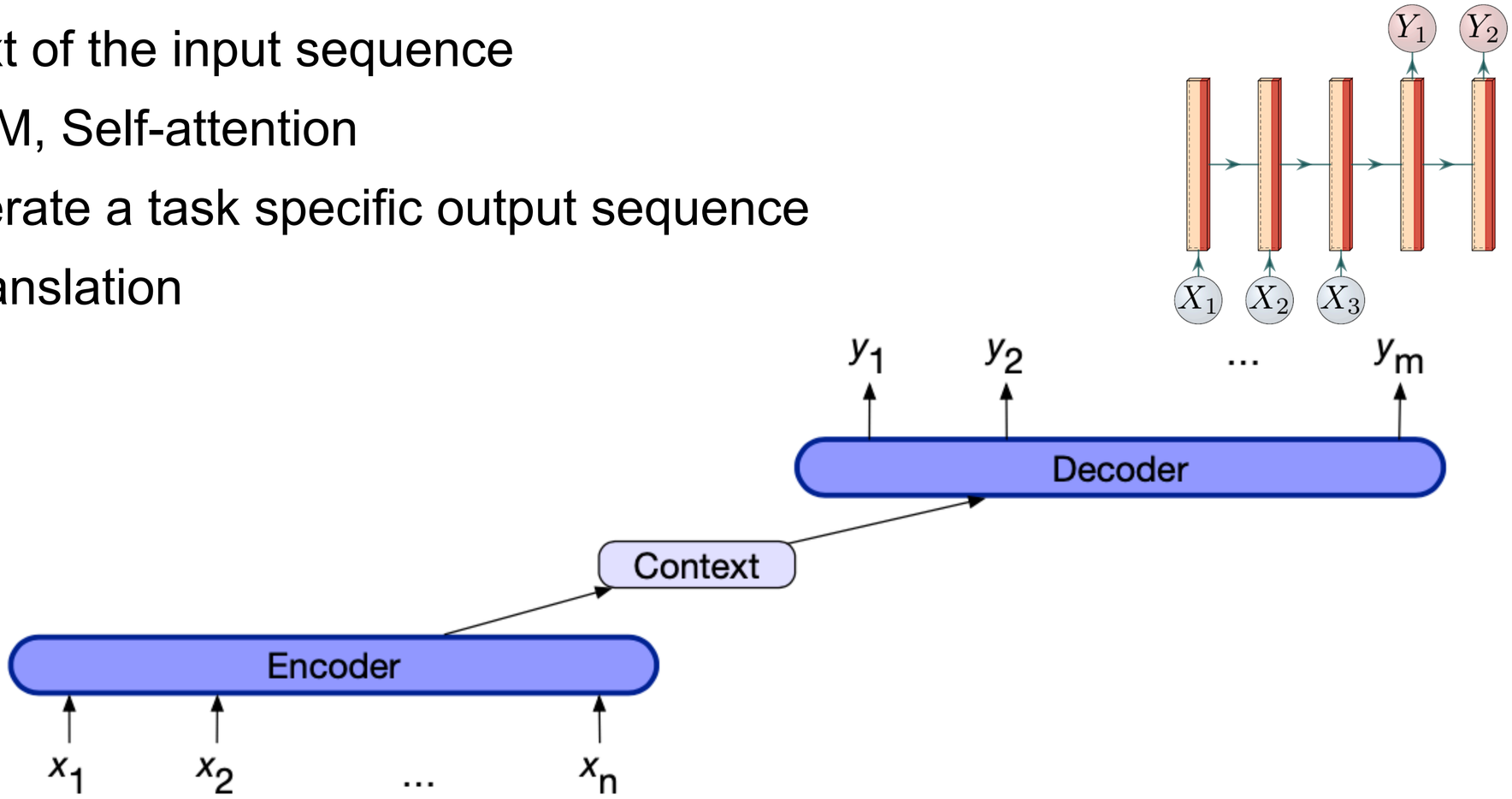
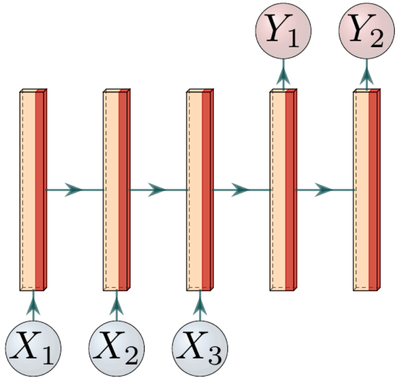


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Encoder-Decoder with RNN

- $p(y) = p(y_1)p(y_2 \mid y_1)p(y_3 \mid y_1, y_2), \cdots, p(y_n \mid y_1, \cdots, y_{n-1})$
  - $h_t = f_h(W_{xh}^T X_t + W_{hh}^T h_{t-1} + b_h)$
  - $Y_t = f_o(W_{yh}^T h_t + b_y)$



- Now if we have a source sequence/text x and the target sequence/text y then
  - $p(y \mid x) = p(y_1 \mid x)p(y_2 \mid y_1, x)p(y_3 \mid y_1, y_2, x), \cdots, p(y_n \mid y_1, \cdots, y_{n-1}, x)$

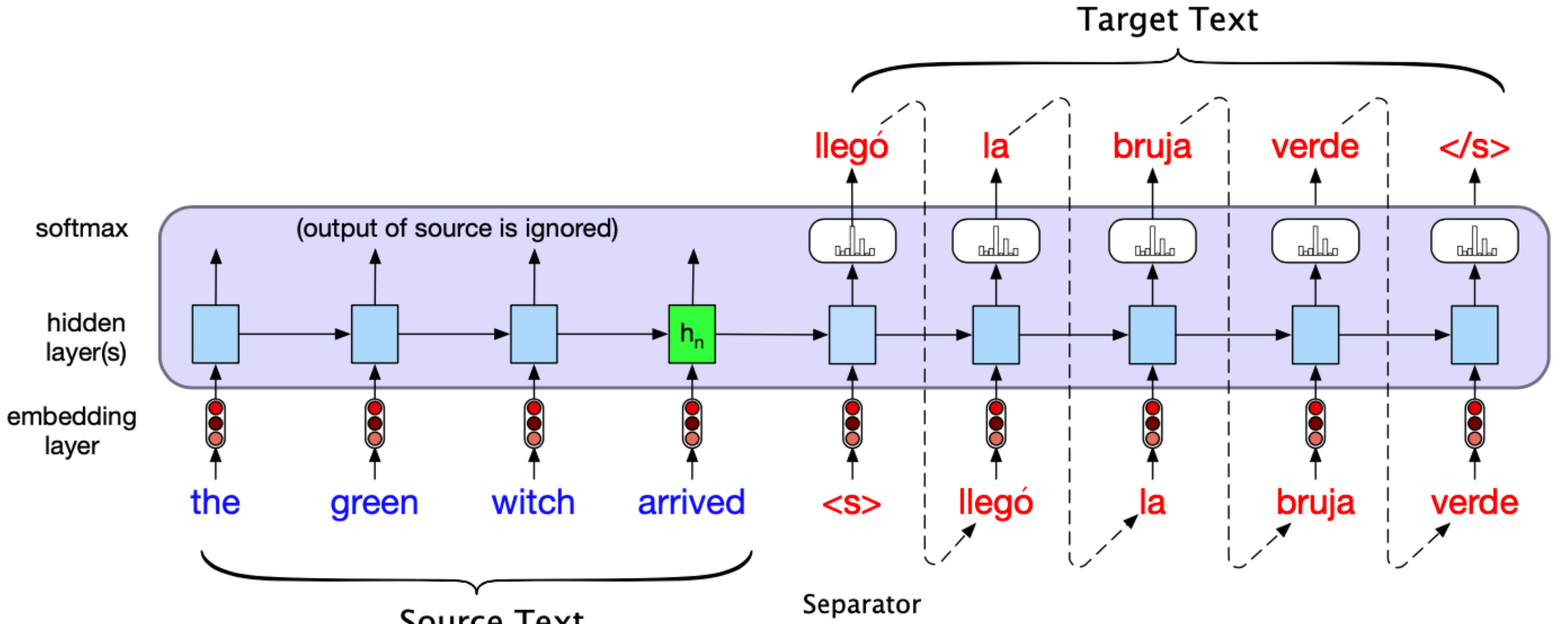
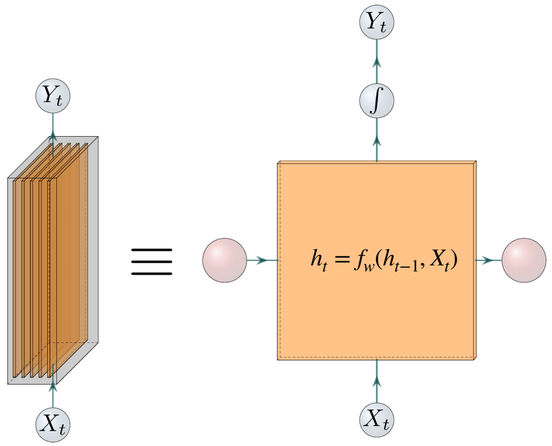
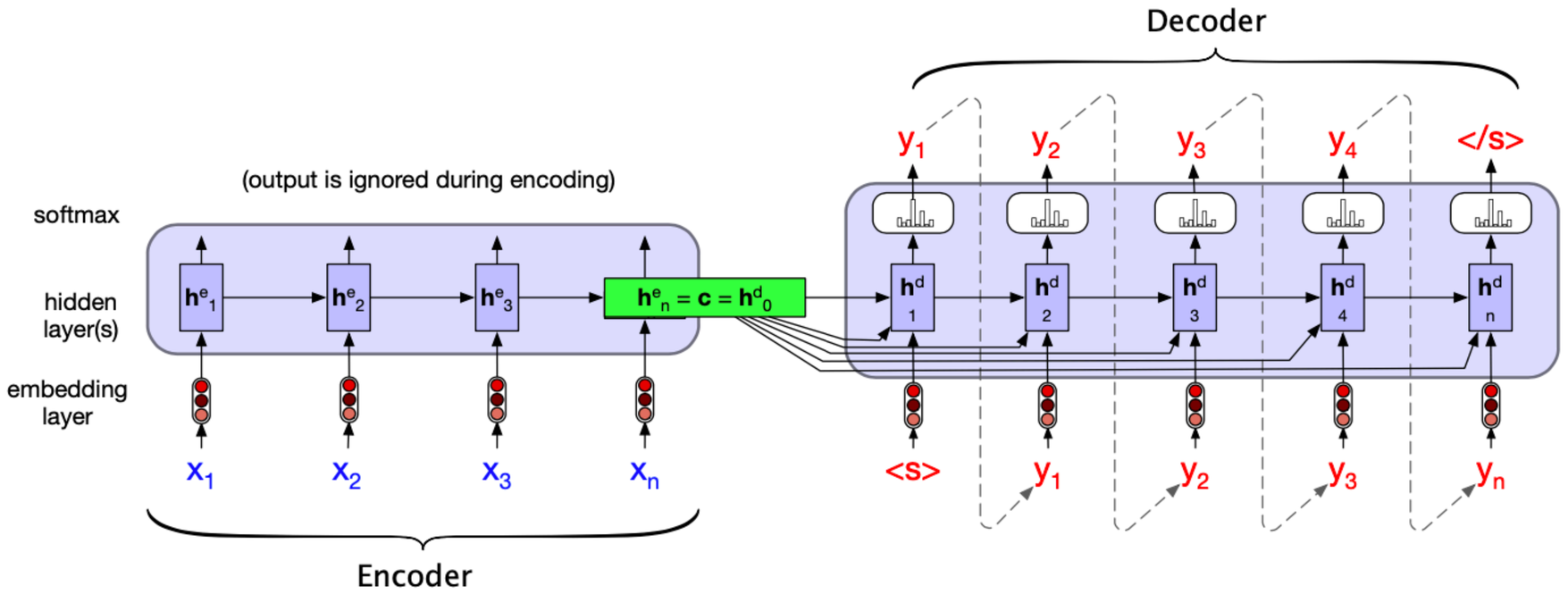


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Encoder-Decoder with RNN (cont.)





- $h_t = f_h(W_{xh}^T X_t + W_{hh}^T h_{t-1} + b_h)$
- $Y_t = f_o(W_{hy}^T h_t + b_y)$

- $h_t^d = f_h^d([W_{xh}^d]^T y_{t-1} + [W_{hh}^d]^T h_{t-1}^d + [W_{ch}^d]^T c + b_h^d)$
- $y_t = f_o^d(W_{yh}^{dT} h_t^d + b_y^d)$
- $c = h_n^e$ and $h_0^d = c$

Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Training a Encoder-Decoder model

- Given a sequence pairs:
  - Source: English
  - Target: Bengali/hindi/German
- Teacher forcing

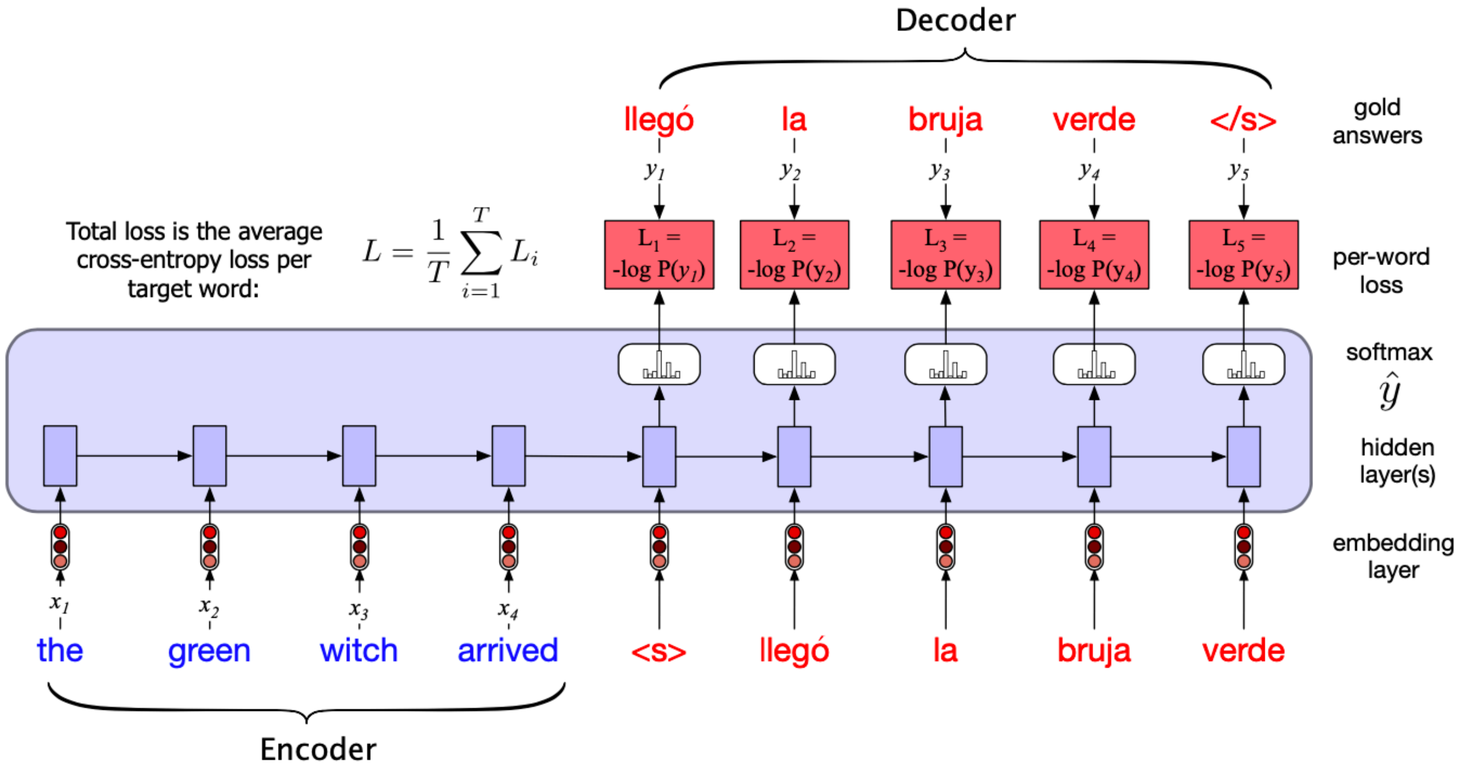
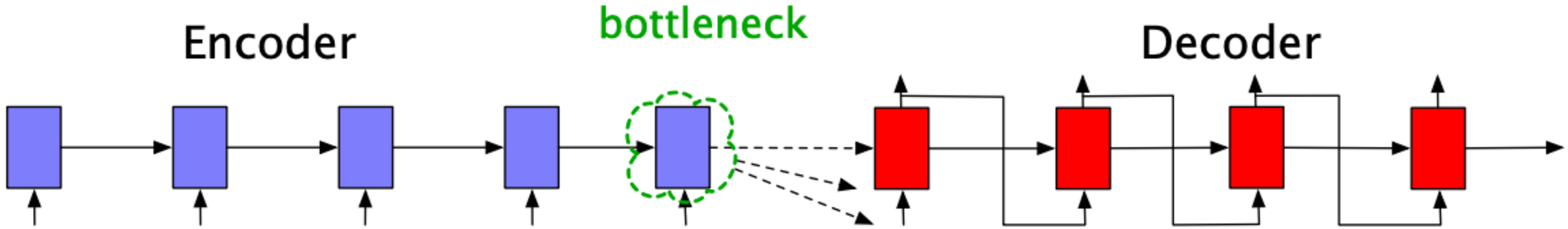


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Can we add attention in encoder-decoder model?

- $h_t^d = f_h^d([W_{xh}^d]^T y_{t-1} + [W_{hh}^d]^T h_{t-1}^d + [W_{ch}^d]^T c + b_h^d)$
- $y_t = f_o^d(W_{yh}^{dT} h_t^d + b_y^d)$
- $c = h_n^e$ and $h_0^d = c$



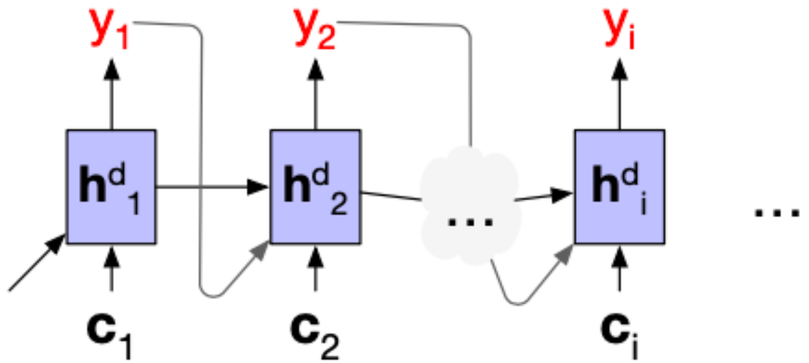


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Adding attention in encoder-decoder model

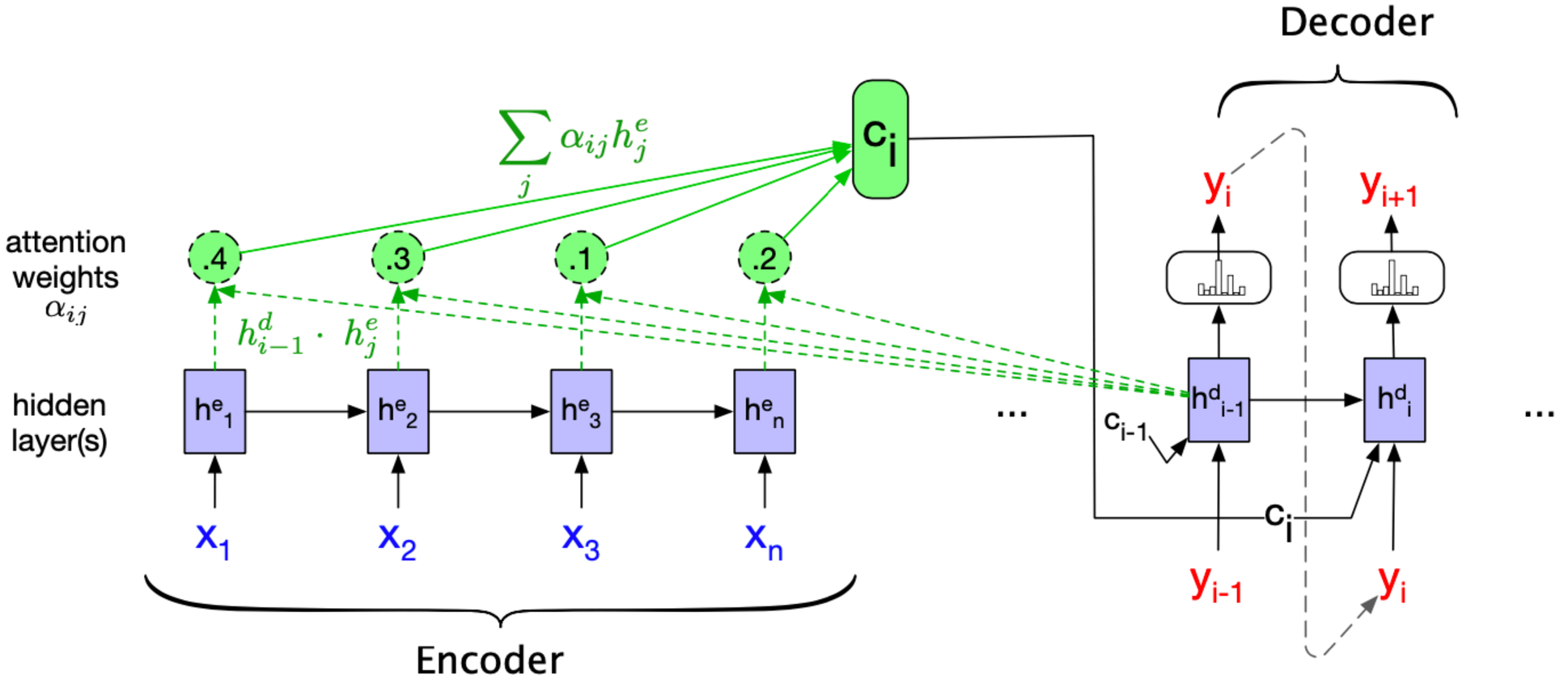
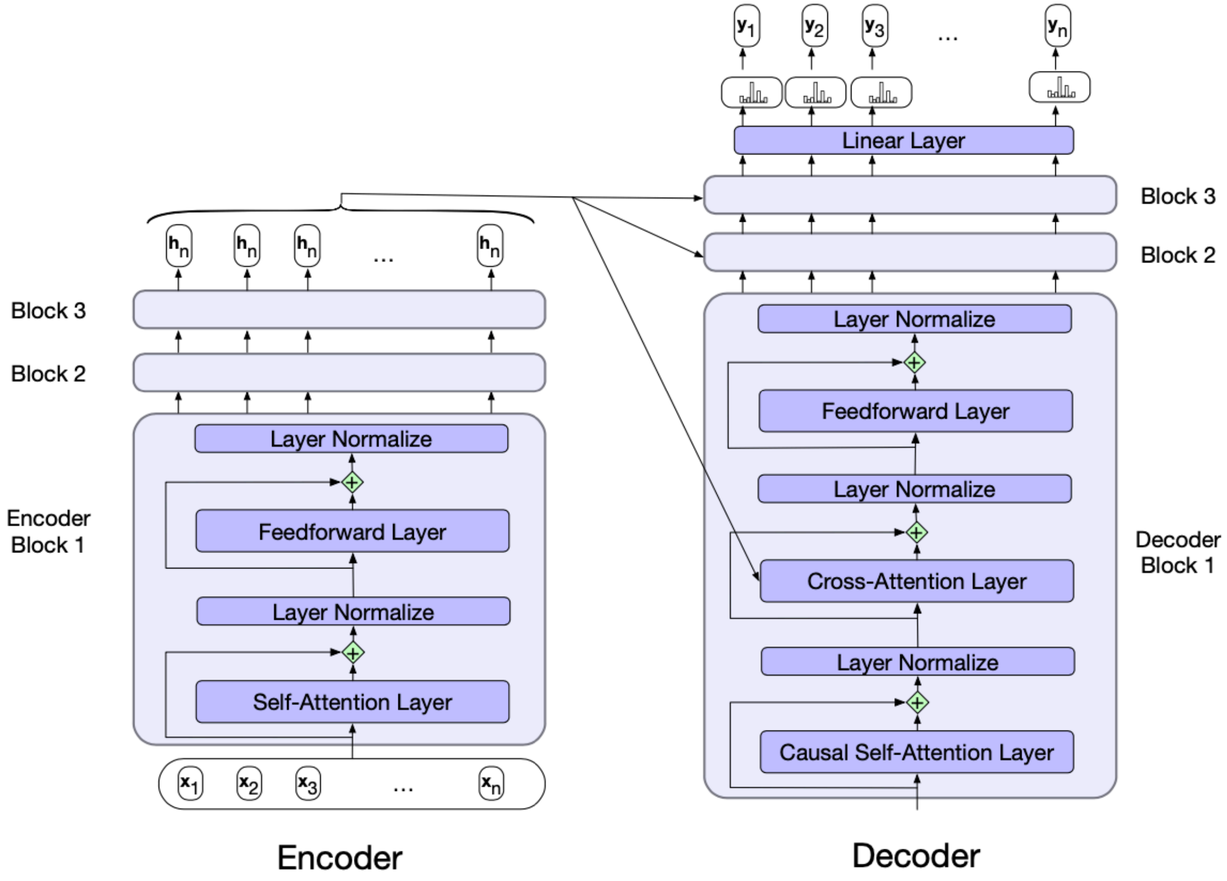


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Encoder-Decoder with Transformer



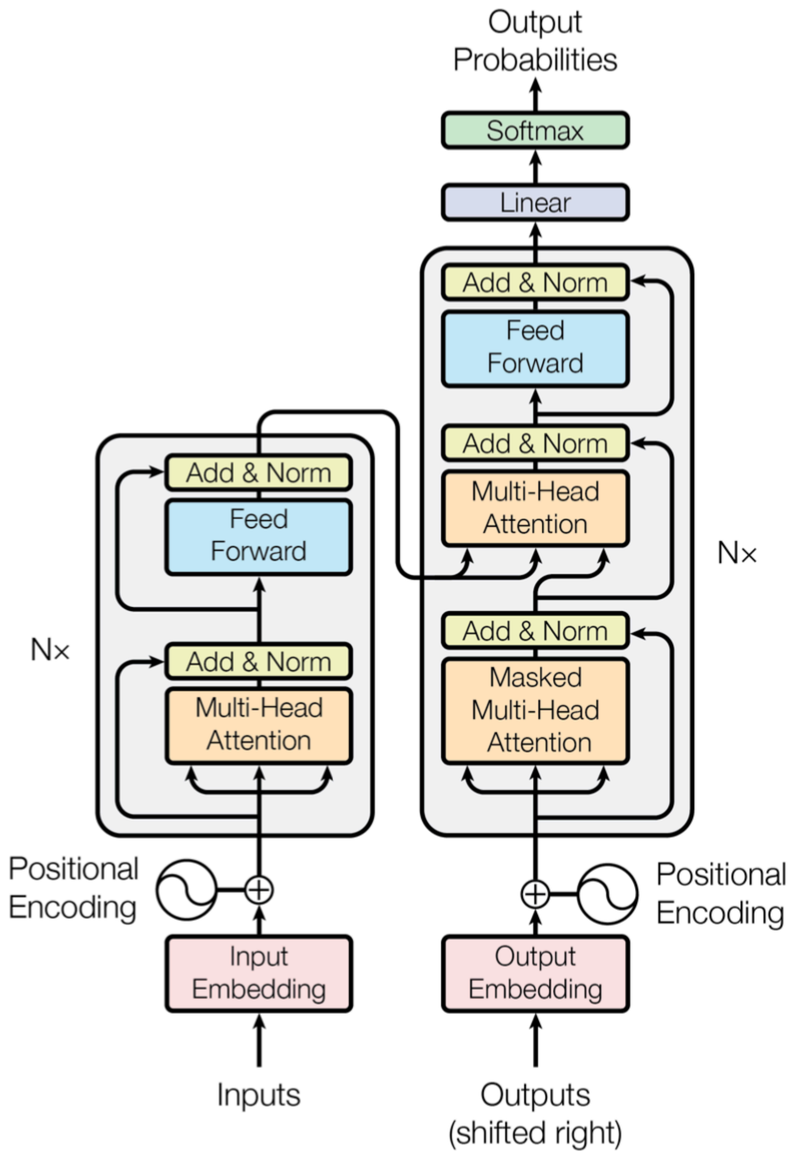


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023

Encoder-Decoder with Transformer

- Cross attention layer:
  - $H^e \in \mathbf{R}^{n \times d}, y_{t-1} \in \mathbf{R}^d$
  - $W^Q \in \mathbf{R}^{d \times d}, W^K \in \mathbf{R}^{d \times d}, W^V \in \mathbf{R}^{d \times d}$
  - $y_t = \left[(y_{t-1}W^Q)(H^eW^K)^T\right](H^eW^V) \in \mathbf{R}^d$

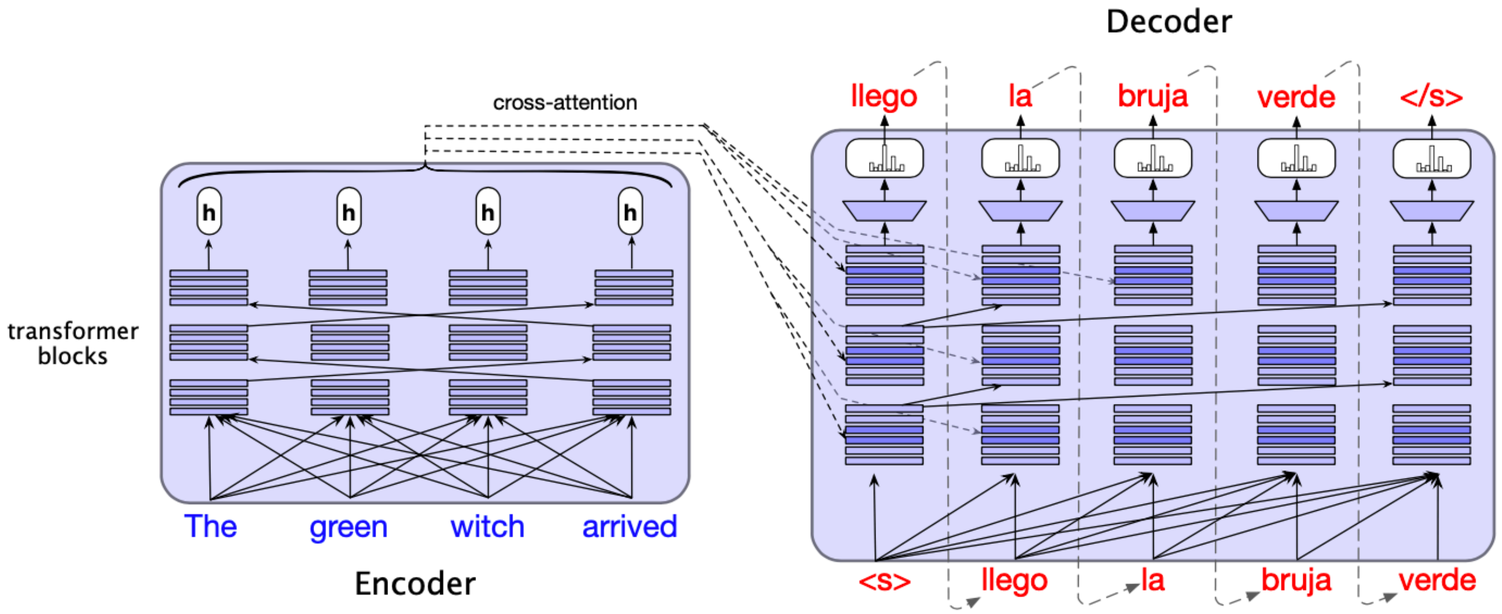


Image: Jurafsky & Martin "Speech and Language Processing, 3rd ed., 2023