Trilha AI / ML

Quando ter atenção é melhor que ter memória?

Lúcio Sanchez Passos
Data Science Manager, Santander

Leonardo Piedade
Solutions Architect, AWS

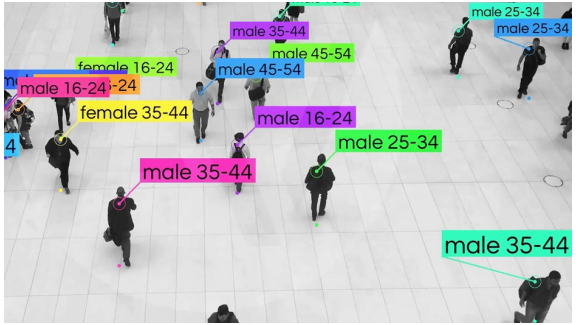
Agenda

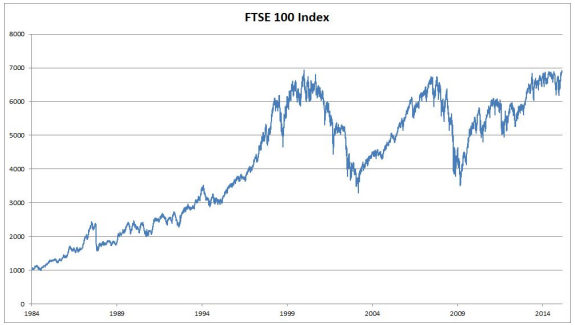
THE DEVELOPER'S CONFERENCE

- Background: Sequences
- Recurrent Process Units: RNN, LSTM and GRU
- Seq2Seq Overview
- Attention & Transformers: How, When, and Why?
- Demo

Sequences - When order matters!

THE DEVELOPER'S CONFERENCE

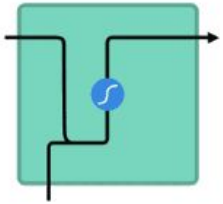




Recurrent Process Units

THE DEVELOPER'S CONFERENCE

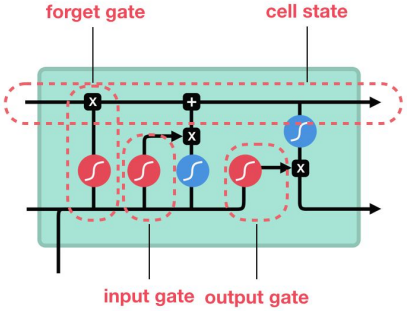
Recurrent Neural Network (RNN)



Good at Modeling Sequence Data

Short-Term Memory Problem

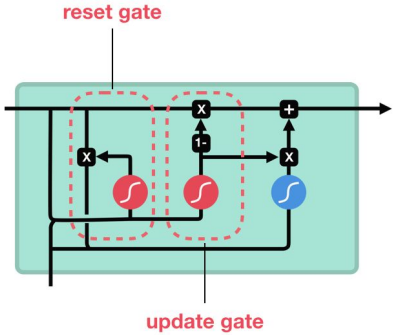
Long Short-Term Memory (LSTM)



Short-Term Memory Problem

More Complex Training Process

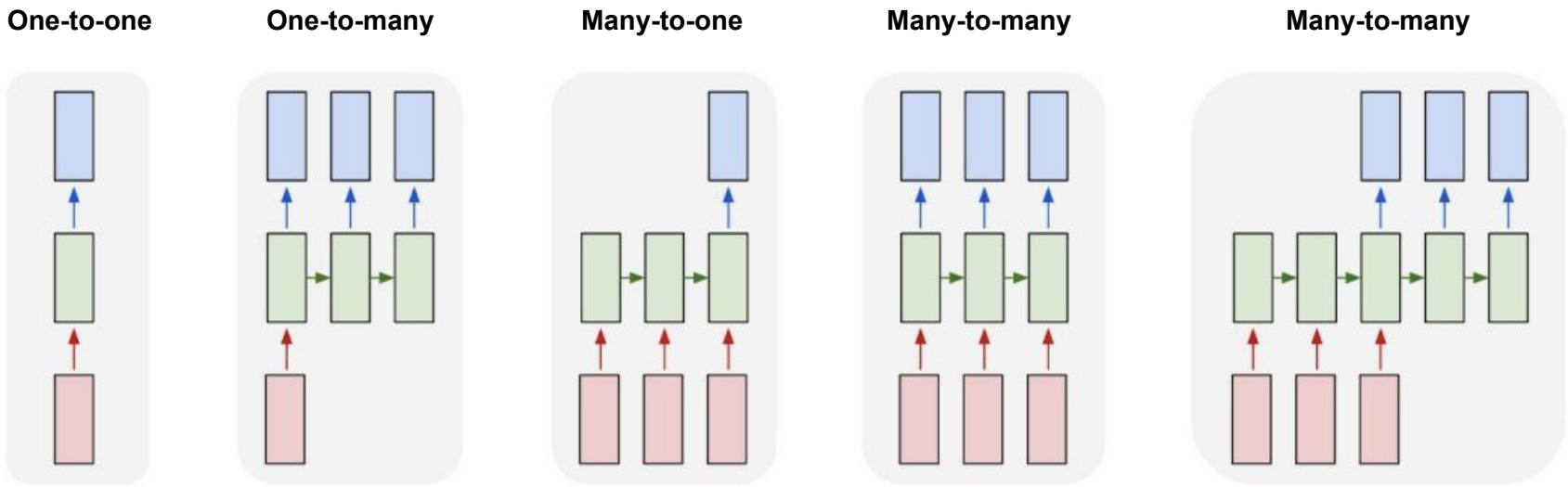
Gated Recurrent Units (GRU)



More Complex Training Process

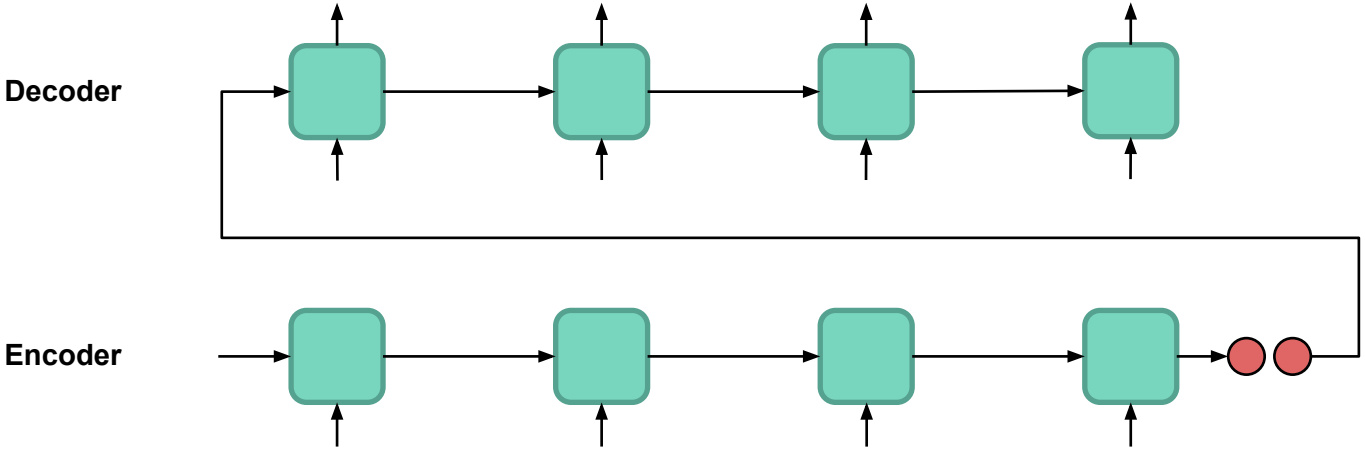
credits: Michael Phi

Applications of RNNs



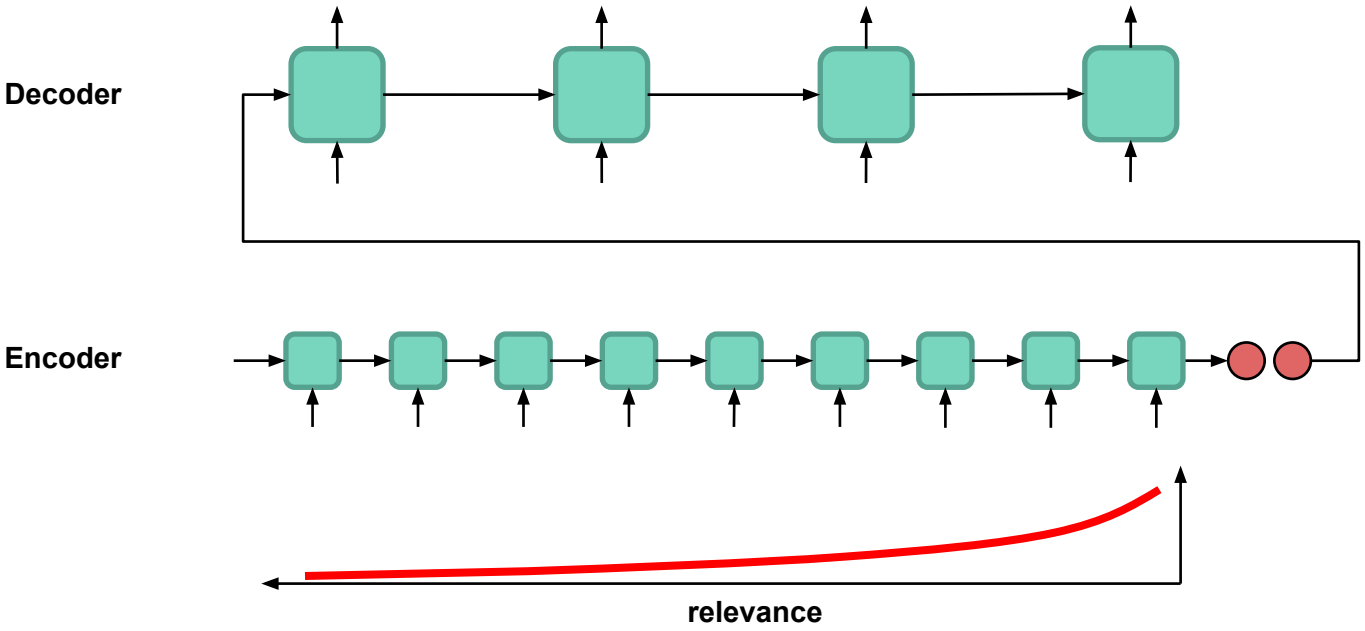
credits: Andrej Karpathy

Seq2Seq (Many-to-many)



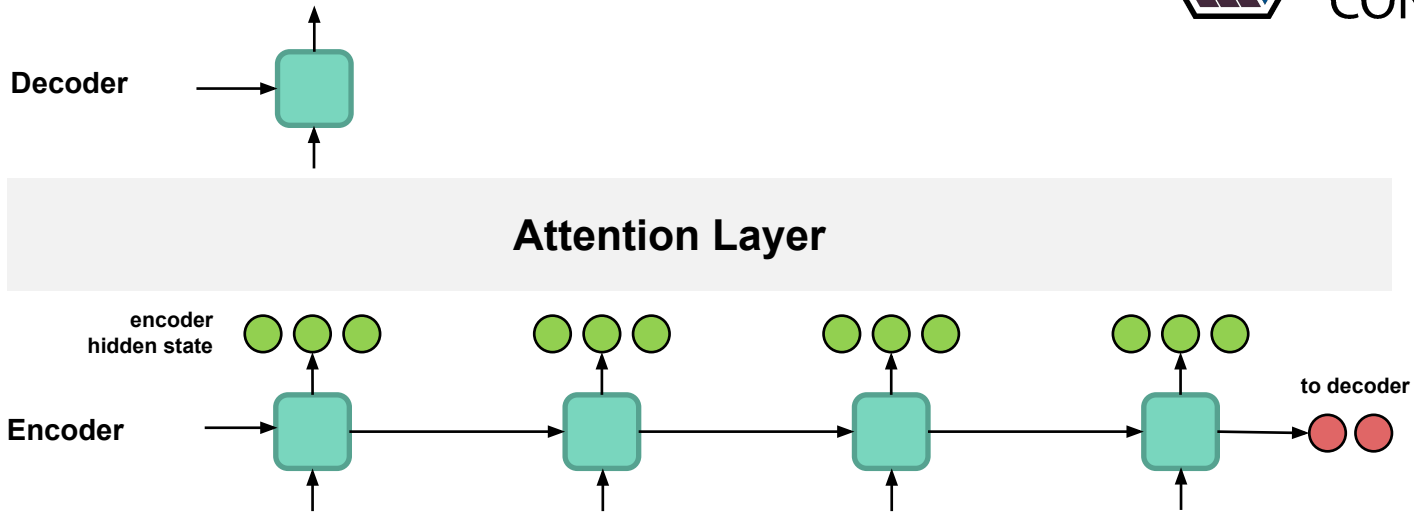
source: [Attn: Illustrated Attention]

Seq2Seq – Bottleneck Problem



source: [Attn: Illustrated Attention]

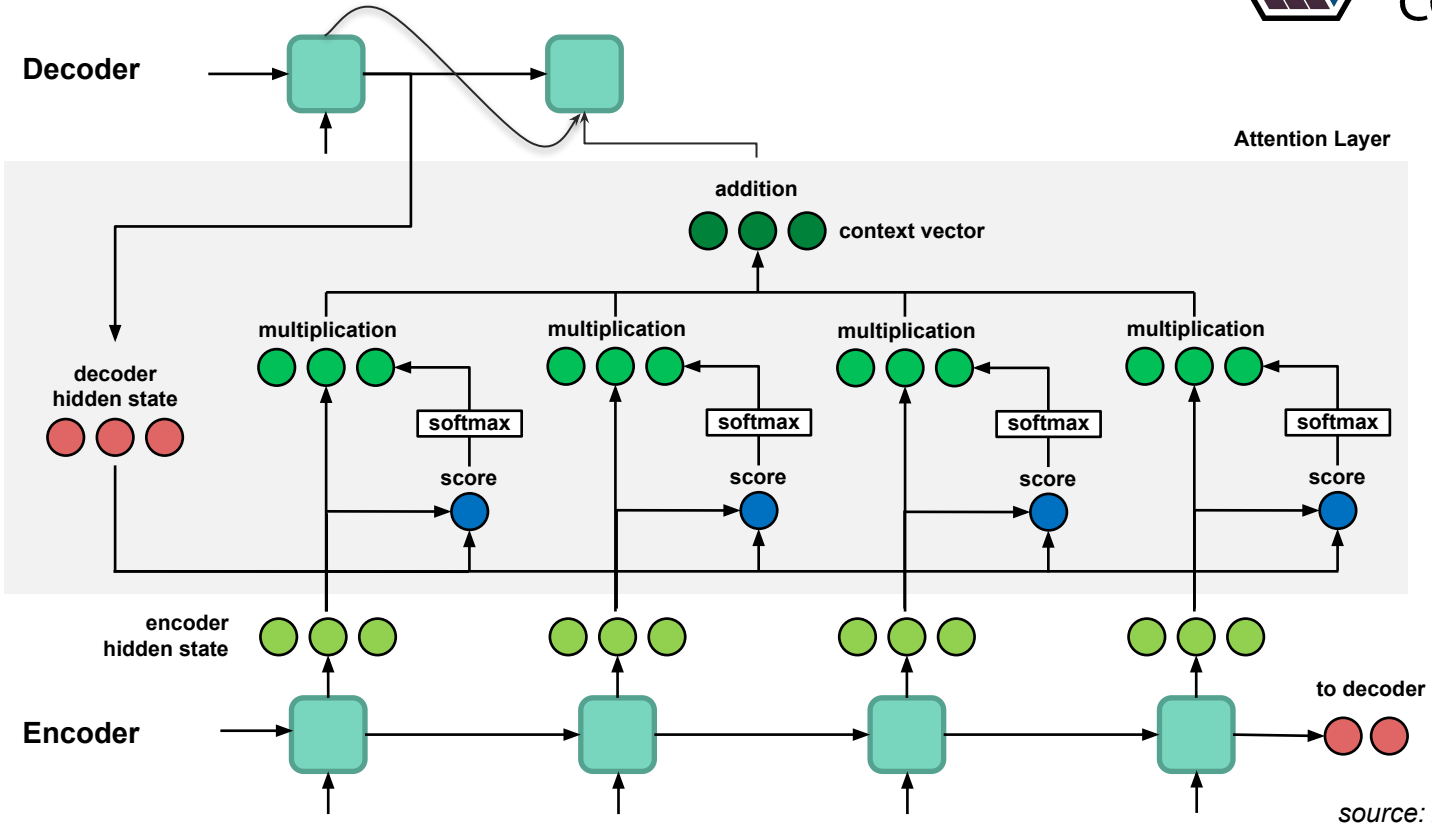
Attention – Definition



"Attention is an interface between the encoder and decoder that provides the decoder with information from every encoder hidden state"

source: [Attn: Illustrated Attention]

Attention Mechanism



source: [Attn: Illustrated Attention]

Attention is great...

THE DEVELOPER'S CONFERENCE

- Attention significantly improves performance (in many applications)
- Attention solves the bottleneck problem
- Attention helps with vanishing gradient problem
- Attention provides some interpretability

credits: Abigail See, Stanford CS224n

Seq2Seq + Attention - Drawback

THE DEVELOPER'S CONFERENCE

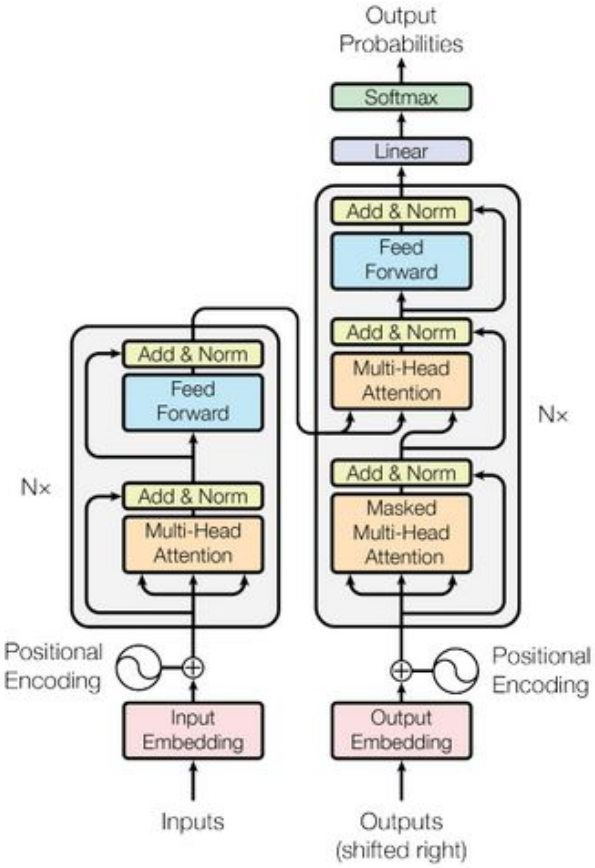
- Sequential computation of data prevents parallelism
- Even with LSTM/GRU + Attention, the gradient vanishing problem is not completely solved

But if we have all states with Attention…why use RNN?

Transformers

THE DEVELOPER'S CONFERENCE

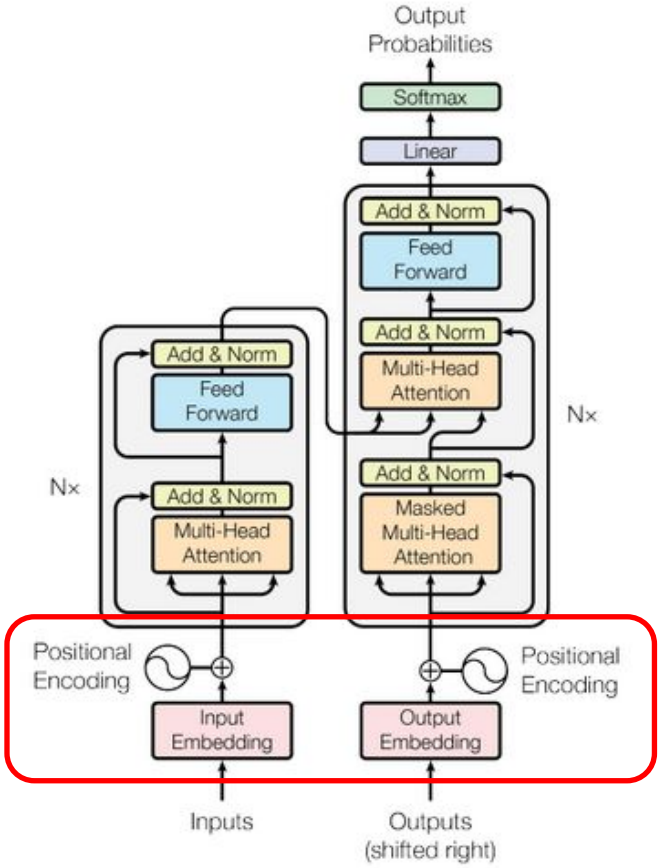
Encoder →



← Decoder

source: https://arxiv.org/pdf/1706.03762.pdf

Transformers

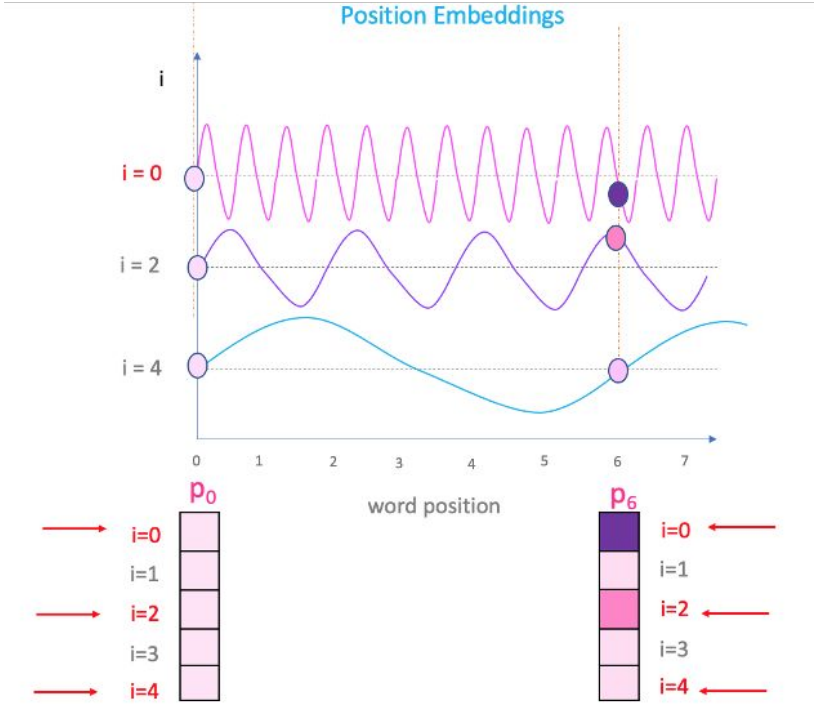


Encoder →

← Decoder

source: https://arxiv.org/pdf/1706.03762.pdf

Positional Encoding



$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}})$$

pos – position in the sequence

d – size of token vector

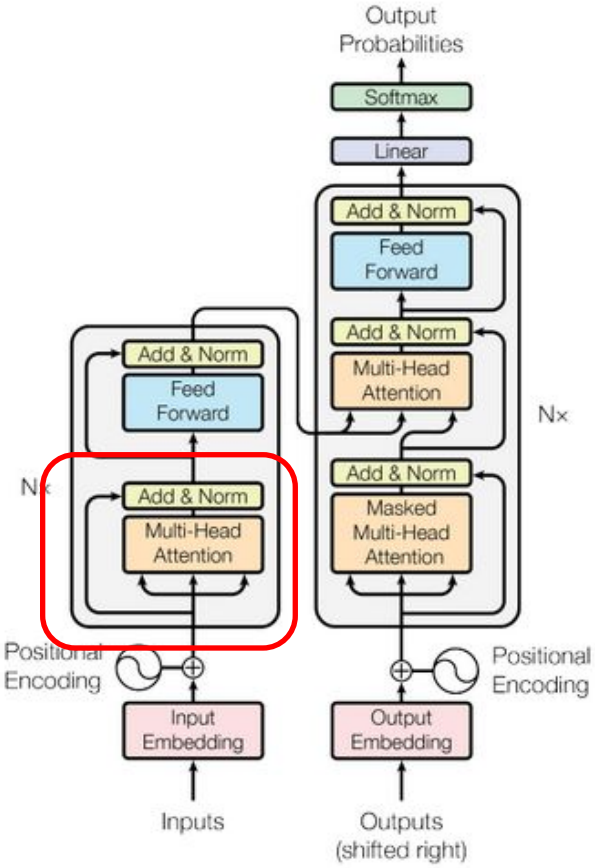
i – position in the tokenized vector

credits: Amirhossein Kazemnejad

Transformers

THE DEVELOPER'S CONFERENCE

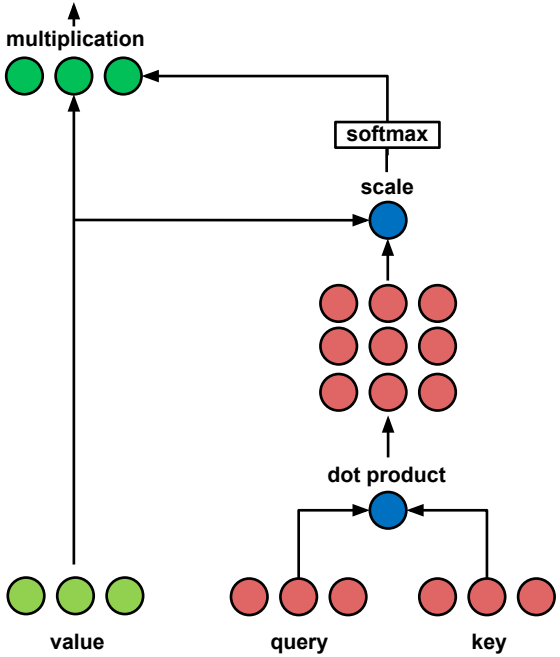
Encoder →



← Decoder

source: https://arxiv.org/pdf/1706.03762.pdf

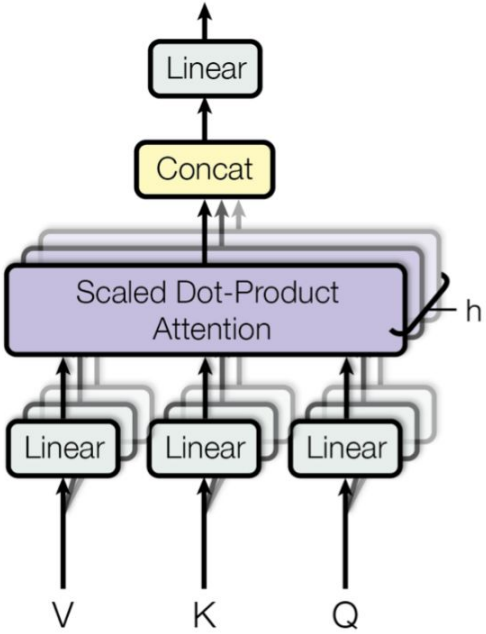
Self-Attention



Self-attention measures the relevance of interaction among all inputs.

source: https://arxiv.org/pdf/1706.03762.pdf

Multi-headed Attention



"Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions."

source: https://arxiv.org/pdf/1706.03762.pdf

Transformers Summary

THE DEVELOPER'S CONFERENCE

- Easier to train (parallel training)
- No gradient vanishing and explosion
- Allows Transfer Learning

THE DEVELOPER'S CONFERENCE

Demo...

Original Papers and Presentations...

THE DEVELOPER'S CONFERENCE

- Attention Is All You Need
- Long Short-Term Memory
- Attn: Illustrated Attention
- Illustrated Guide to Transformers
- Attentional Neural Network Model
- Transcoder: Facebook's Unsupervised Programming Language Translator

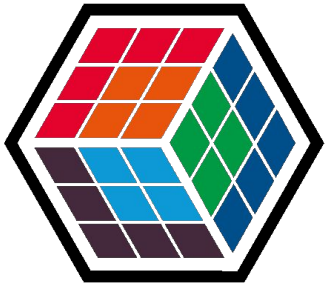
Quando ter atenção é melhor que ter memória?

THE DEVELOPER'S CONFERENCE

Obrigado!

linkedin.com/in/luciopassos/

linkedin.com/in/leoap/



THE DEVELOPER'S CONFERENCE