Embeddings, self-attention, recursion

Is this all we need?

Filippo Biscarini
Senior Scientist
CNR, Milan (Italy)

Nelson Nazzicari
Senior Scientist
CREA, Lodi (Italy)

All you need is love?

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

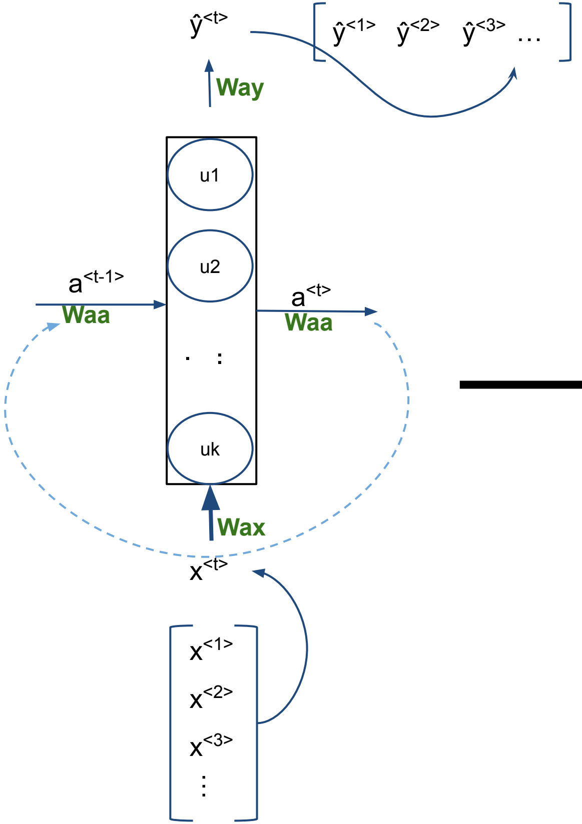
Aidan N. Gomez[*, †]
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser[*]
Google Brain
lukaszkaiser@google.com

Illia Polosukhin[*, ‡]
illia.polosukhin@gmail.com

12/06/2017

From RNN to transformers



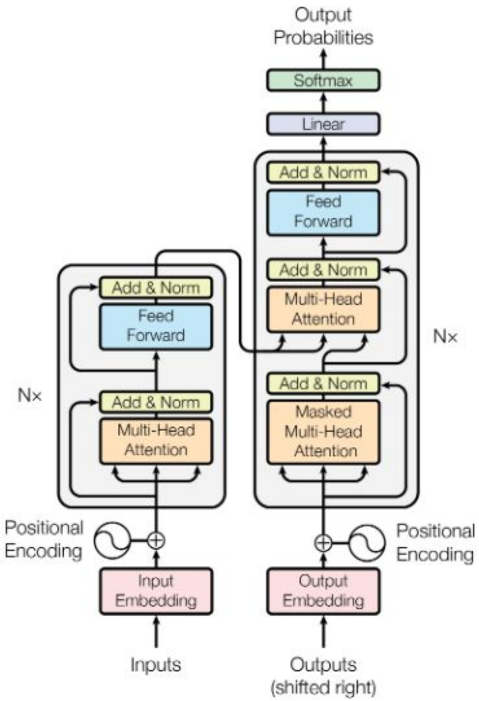
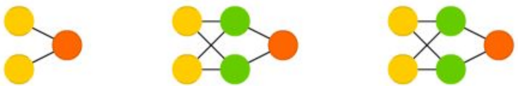


Figure 1: The Transformer - model architecture.



- RNN, LSTM, GRU etc.: sequential calculations, <u>no parallelization possible</u> (severe computational limit)
- transformers capture long-range dependencies in the data and at the same time are <u>amenable to parallelization</u> (major computational advantage)

From RNN to transformers

- transformers are a new network architecture that **dispenses with recurrence and convolutions entirely** (no CNN, no RNN)
- attention is the engine of **transformer models**
- **(self)attention**: ability of the model to automatically, dynamically and independently **highlight** and **use** the **salient parts of the input data**
- transformers are successfully <u>applied also to image data</u> (computer vision)

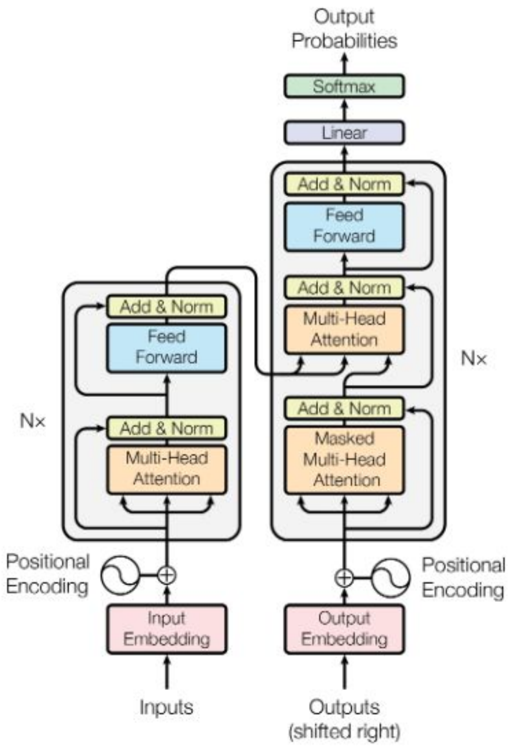
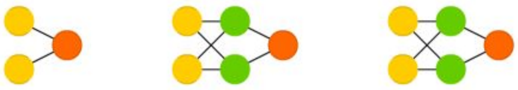


Figure 1: The Transformer - model architecture.



From RNN to transformers

- transformers are a new network architecture that **dispenses with recurrence and convolutions entirely** (no CNN, no RNN)
- attention is the engine of **transformer models**
- **(self)attention**: ability of the model to automatically, dynamically and independently **highlight** and **use** the **salient parts of the input data**
- transformers are successfully <u>applied also to image data</u> (computer vision)

attention?

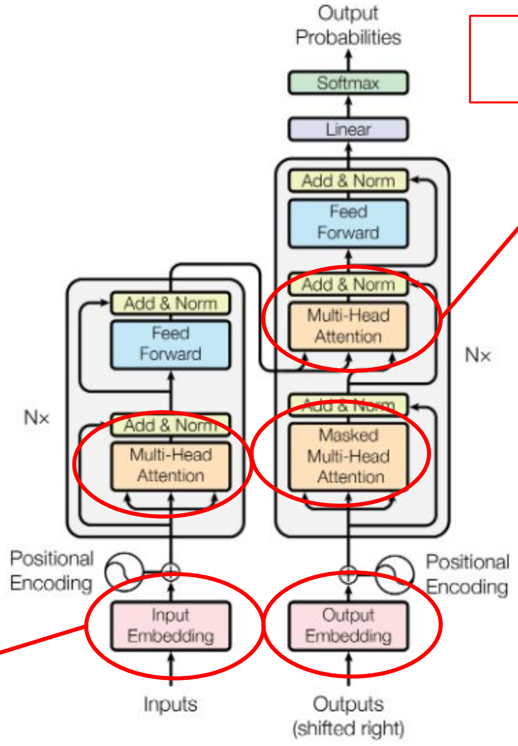
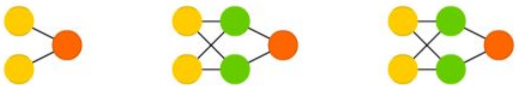


Figure 1: The Transformer - model architecture.

Embeddings and attention are not specific to transformers, but transformers make heavy use of them (and in specific ways)

embeddings?



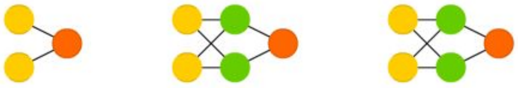
Embeddings (text)

- From NLP: mainly text embeddings (but also: sounds, videos, images etc.)
- Data (text) is projected into a <u>multi-dimensional latent space</u>
- large sparse high-dimensional data → **dense lower dimensional representation**
- **manifold** (sort of “multidimensional set”): similar items are close to one another (e.g. sentences that are semantically similar should have similar embedded vectors)
- distances between data (e.g. words, sentences, images etc.) are calculated based on the “new features” (embeddings) in the high-dimensional latent space
- e.g. cosine distance: $\cos\_d(\mathbf{u},\mathbf{v}) = 1 - (\mathbf{u} \cdot \mathbf{v}) / (||\mathbf{u}|| * ||\mathbf{v}||)$

- The DNN model will learn embeddings so to preserve distances between similar items

1	0		0	1
0	0	...	1	0
0	2		0	0
...	...	...	...	...
0	1		0	10
0	0	...	0	0
1	0		0	0

0.235	0.501		0.056
-0.179	0.114	...	-0.987
1.993	-0.782		1.002

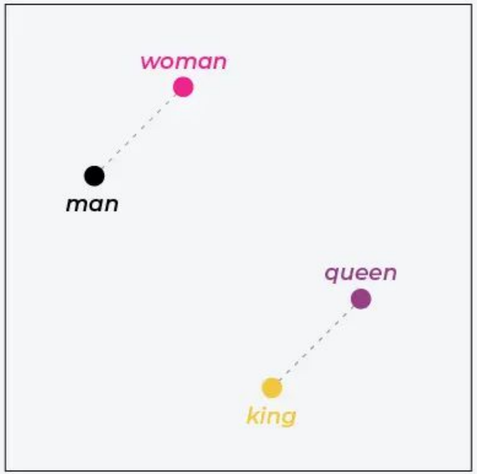


Embeddings (text)

word	living being	feline	human	gender	royalty	verb	plural
man	0.6	-0.2	0.8	0.9	-0.1	-0.9	-0.7
woman	0.7	0.3	0.8	-0.7	0.1	-0.5	-0.4
king	0.5	-0.4	0.7	0.8	0.9	-0.7	-0.6
queen	0.8	-0.1	0.8	-0.9	0.8	-0.5	-0.9

word

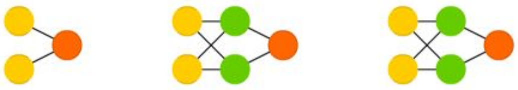
Word embedding



Visualization of word embedding

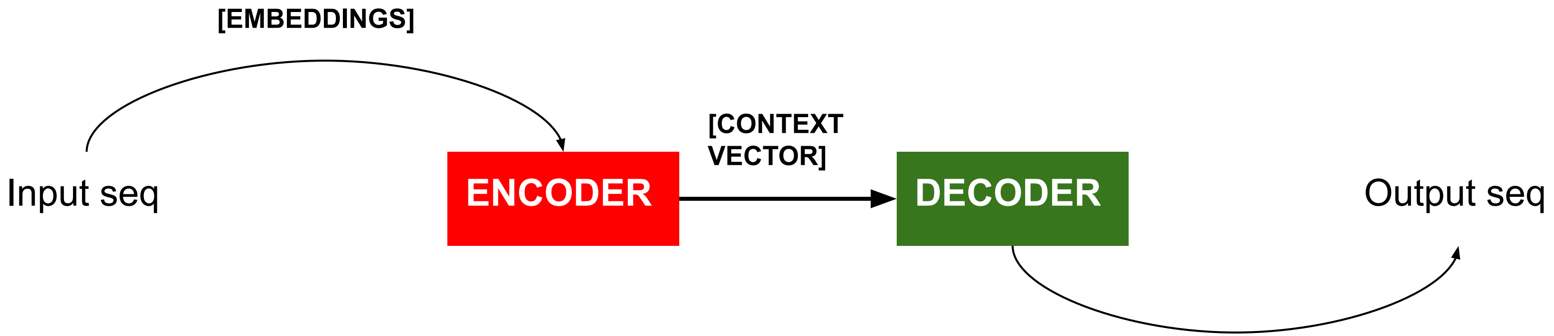
(fantasy example)

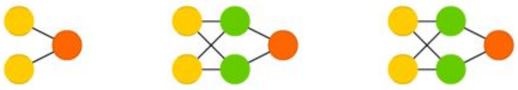
From: https://arize.com/blog-course/embeddings-meaning-examples-and-how-to-compute/



Attention, please!

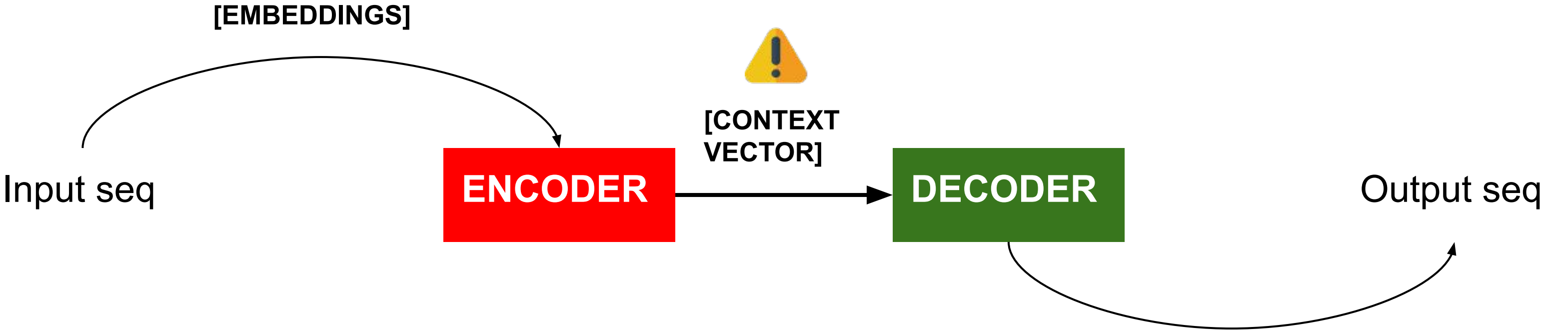
seq2seq models





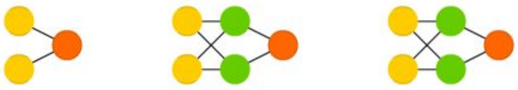
Attention, please!

seq2seq models



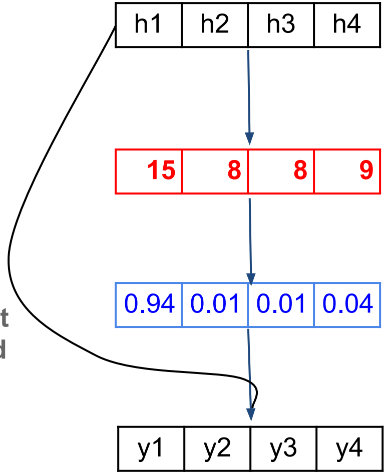
LIMITATIONS

- difficulties with long-range dependencies
- no parallelization



Attention, please!

attention



[multiply input by normalized scores]

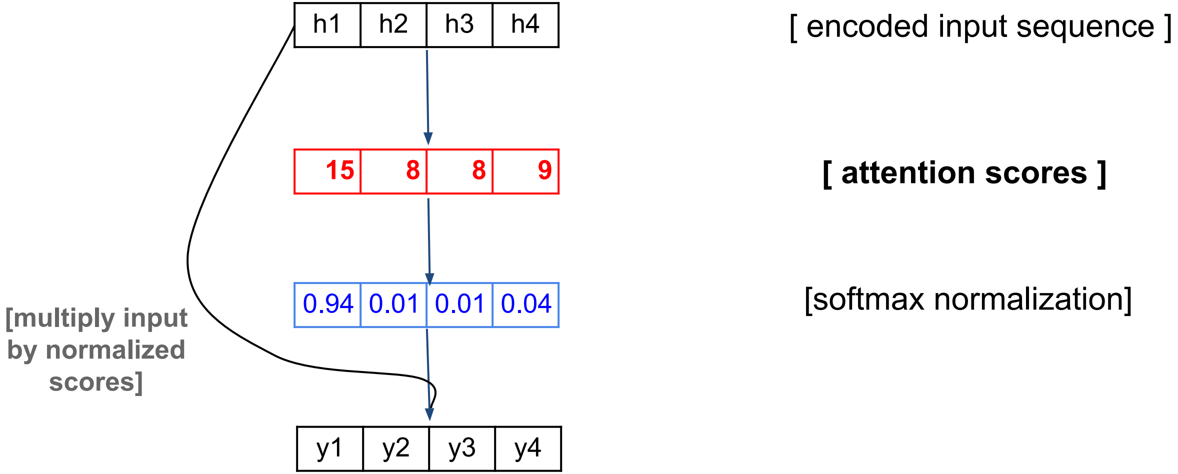
[encoded input sequence]

[attention scores]

[softmax normalization]

Attention, please!

attention



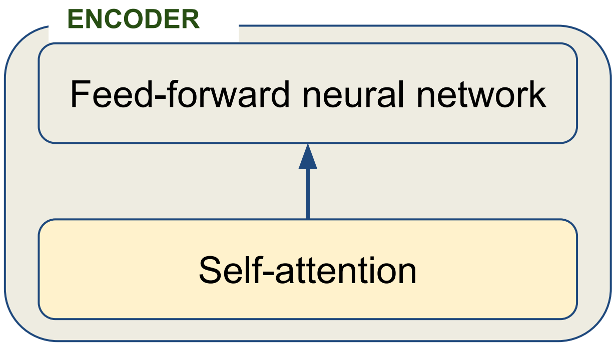
- this takes care of long-range dependencies → better models!
- (the actual mathematical details are much more complex)

(self) attention, please!

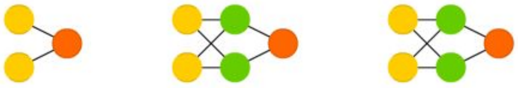
Attention: encoded input → attention scores → decoded output

VS

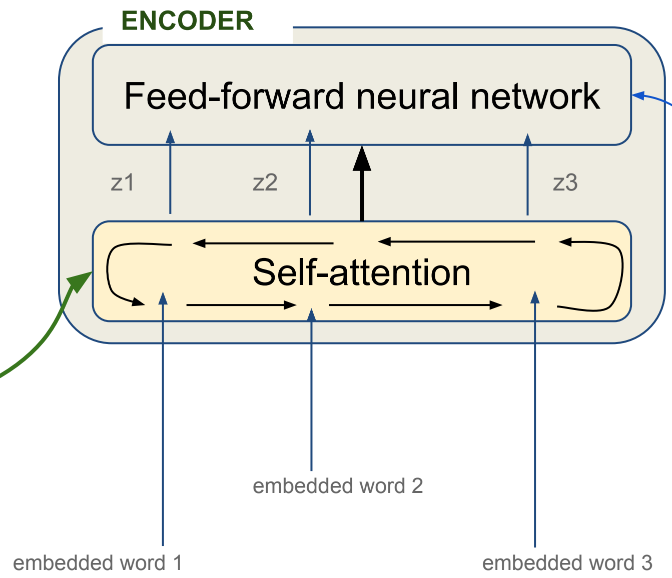
Self attention: (attention + feed forward NN) → encoded input



This is the (basic) transformer’s encoder!

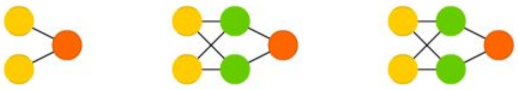


(self) attention, please!



1. this takes into account dependencies between words (elements in the sequence)

2. here elements are independent from each other → **PARALLELIZATION** (by design)



(self) attention, please!

- The parallel processing of elements of the sequence might lead to problems with the reconstruction of the output sequence
- <u>Positional encoding</u>: ensures that the order of the sequence is retained in the model

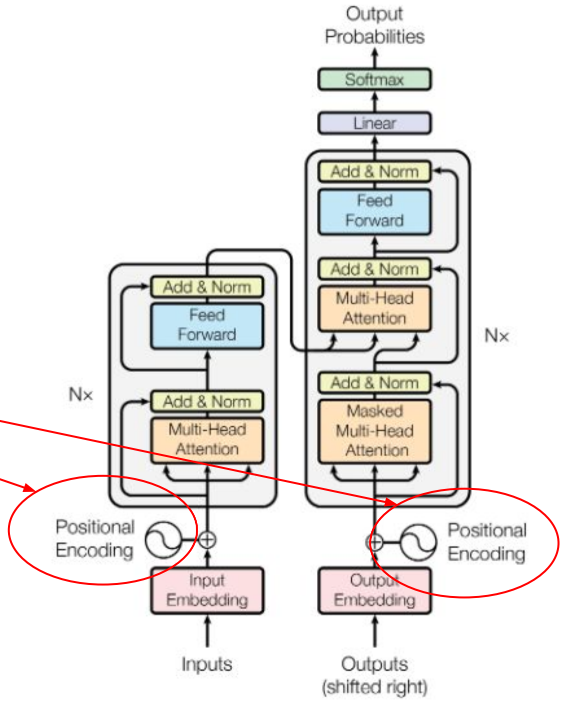
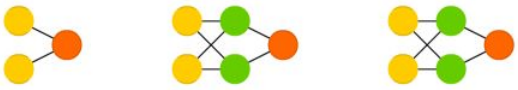


Figure 1: The Transformer - model architecture.



(self) attention, please!

https://jalammar.github.io/illustrated-transformer/

Transformers?

Were RNNs All We Needed?

Leo Feng
Mila – Université de Montréal & Borealis AI
leo.feng@mila.quebec

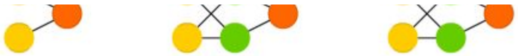
Frederick Tung
Borealis AI
frederick.tung@borealisai.com

Mohamed Osama Ahmed
Borealis AI
mohamed.o.ahmed@borealisai.com

Yoshua Bengio
Mila – Université de Montréal
yoshua.bengio@mila.quebec

Hossein Hajimirsadeghi
Borealis AI
hossein.hajimirsadeghi@borealisai.com

04/10/2024



Were RNNs all we needed?

- transformers (2017) reshaped deep learning: sequence modelling and more
- scalability limitations -particularly with respect to sequence length
- renewed interest in novel RNN models: parallelizable, comparable performance, scale
- LSTMs (1997) and GRUs (2014): by simplifying these models, we can derive minimal versions (**minLSTMs** and **minGRUs**):
    a. use fewer parameters than their traditional counterparts
    b. are parallelizable
    c. competitive performance, rivalling Transformers