Attention Is All You Need

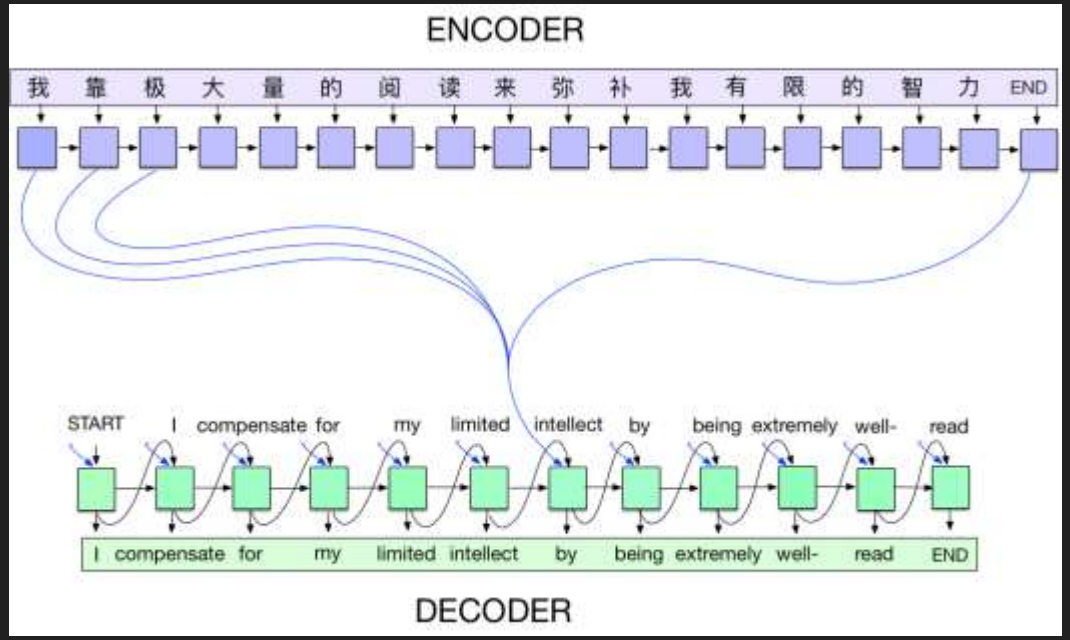
Presenter: Illia Polosukhin, NEAR.ai

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

Work performed while at Google

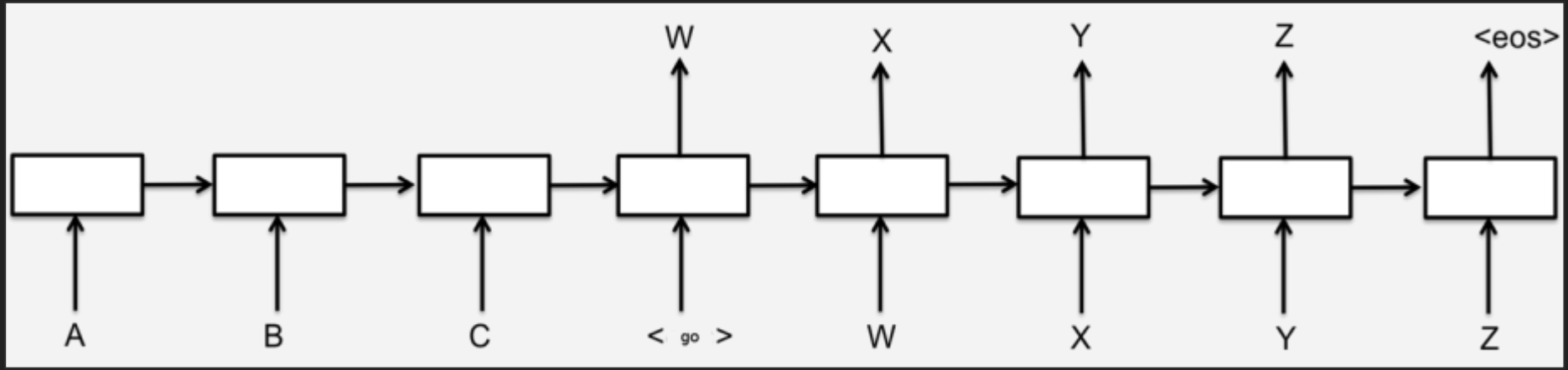
Deep Learning for NLP

- RNNs have transformed NLP
- State-of-the-art across many tasks
- Translation has been recent example of a large win



https://research.googleblog.com/2017/04/introducing-tf-seq2seq-open-source.html

Sequence To Sequence



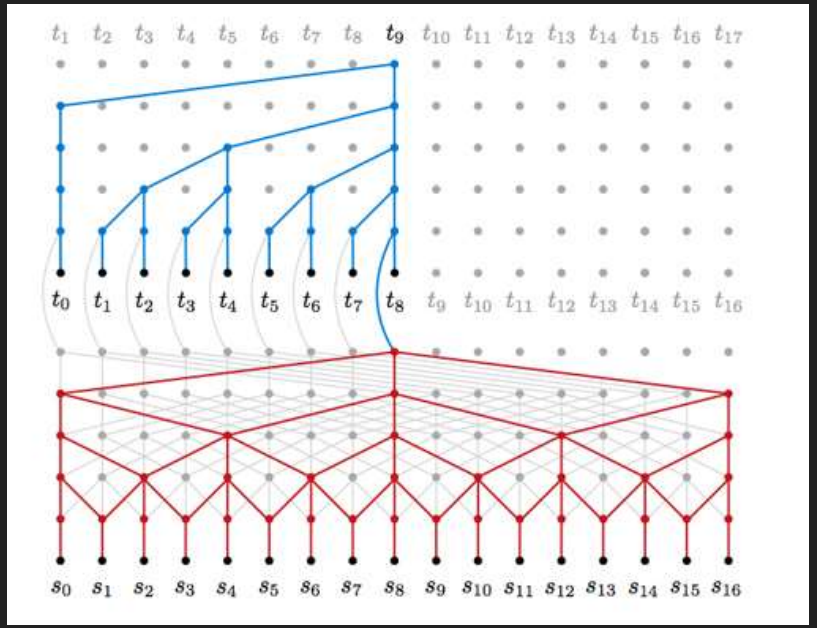
“Sequence to Sequence Learning with Neural Networks”, Sutskever et al. 2014

Problem with RNNs

- Hard to parallelization efficiently
- Back propagation through sequence
- Transmitting local and global information through one bottleneck [hidden state]

Convolutional Models

- Trying to solve the problems with Sequence models
- Notable work:
  - Neural GPU
  - ByteNet
  - ConvS2S
- Limited by size of convolution



Neural Machine Translation in Linear Time, Kalchbrenner et al.

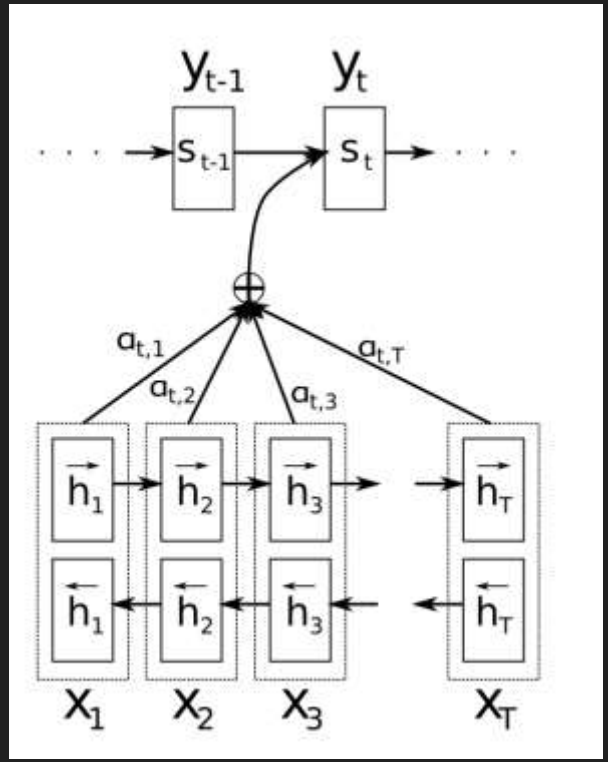
Attention Mechanics

- Removes bottleneck of Encoder-Decoder model
- Provides context for given decoder step

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j.$$

$$\alpha_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{T_x} \exp\left(e_{ik}\right)},$$

$$e_{ij} = a(s_{i-1}, h_j)$$



"Neural Machine Translation by Jointly Learning to Align and Translate", Bahdanau et al.

Self/Intra/Inner Attention in Literature

- “Inner Attention based Recurrent Neural Networks for Answer Selection”, ACL 2016, Wang et al.
- “Learning Natural Language Inference using Bidirectional LSTM model and Inner-Attention”, 2016, Liu et al.
- “Long Short-Term Memory-Networks for Machine Reading”, EMNLP 2016, Cheng et al.
- “A Decomposable Atte… Language Inference”, EMNLP 2016, Parikh et al.

$$f_{ij} := F_{\text{intra}}(a_i)^T F_{\text{intra}}(a_j)\,,$$

$$a'_i := \sum_{j=1}^{\ell_a} \frac{\exp(f_{ij} + d_{i-j})}{\sum_{k=1}^{\ell_a} \exp(f_{ik} + d_{i-k})} a_j\,.$$

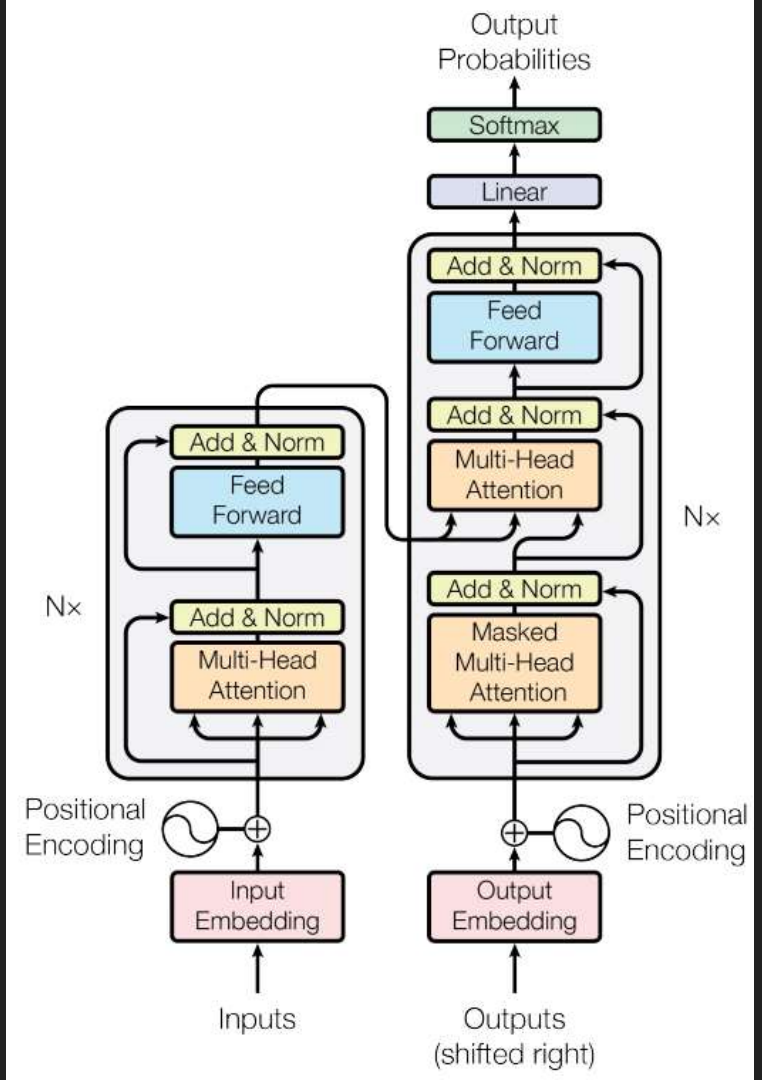
Why self attention?

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

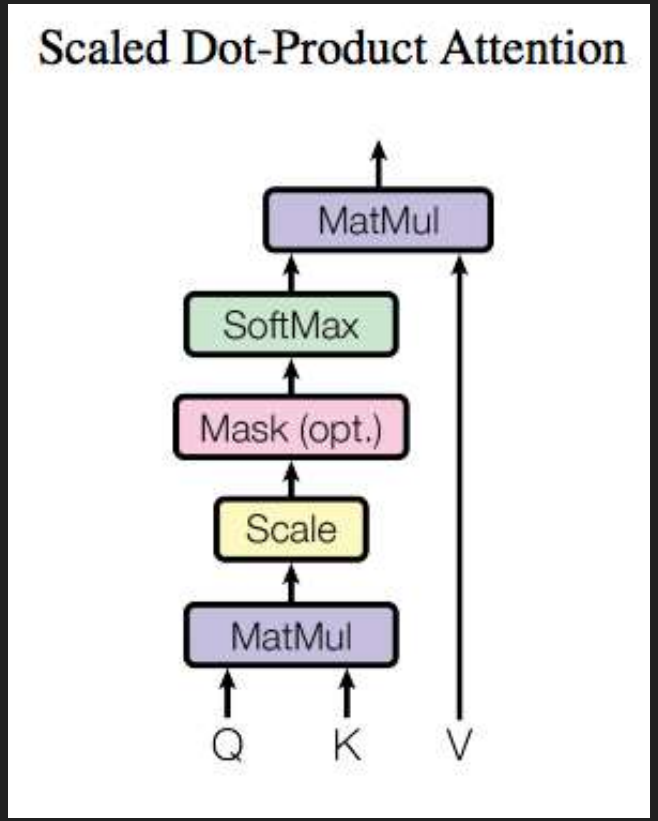
Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

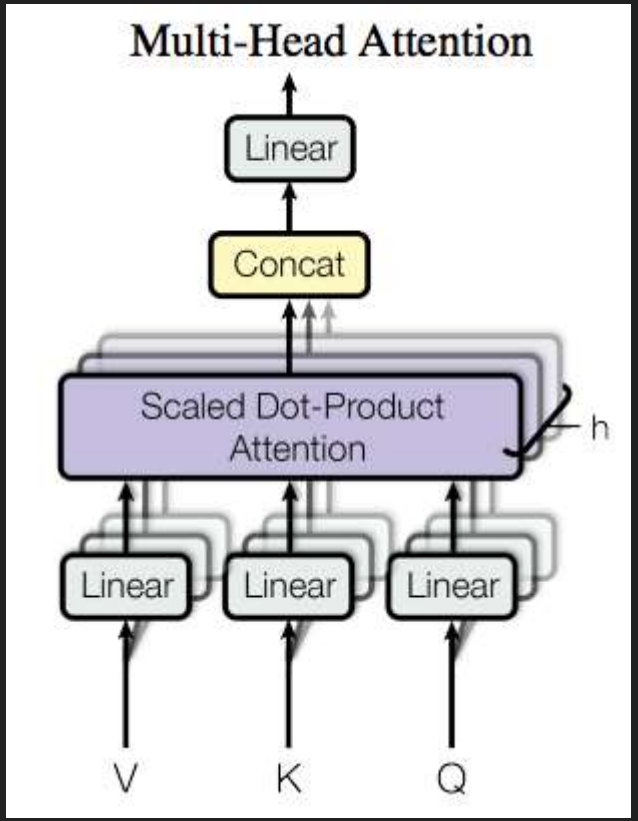
Transformer architecture

- Encoder: 6 layers of self-attention + feed-forward network
- Decoder: 6 layers of masked self-attention and output of encoder + feed-forward.



Scaled Dot Product and Multi-Head Attention





Positional Encoding

- Positional encoding provides relative or absolute position of given token
- Many options to select positional encoding, in this work:

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

Fixed offset PE_{pos+k} can be represented as linear function of PE_{pos}

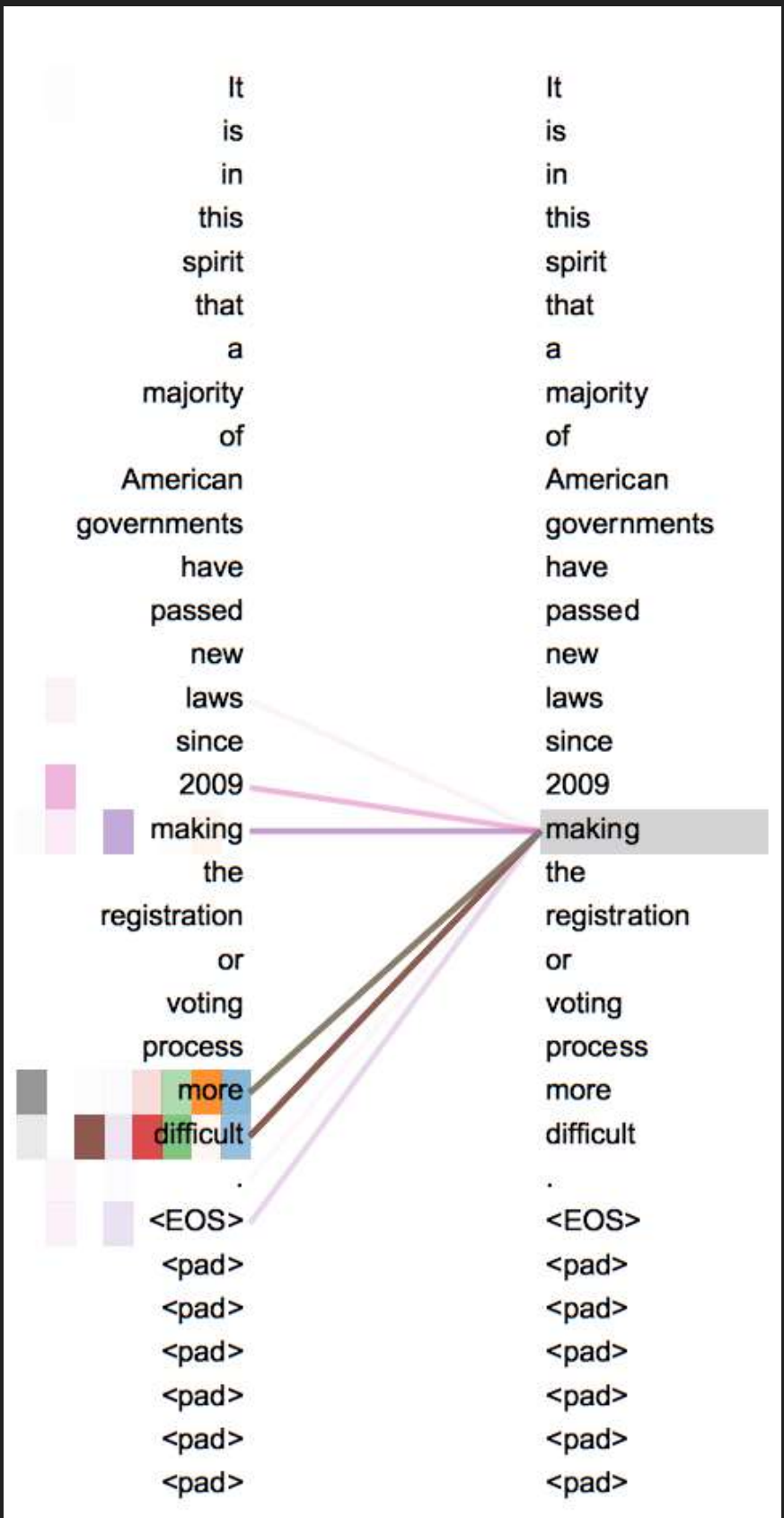
- Alternative, to learn positional embeddings

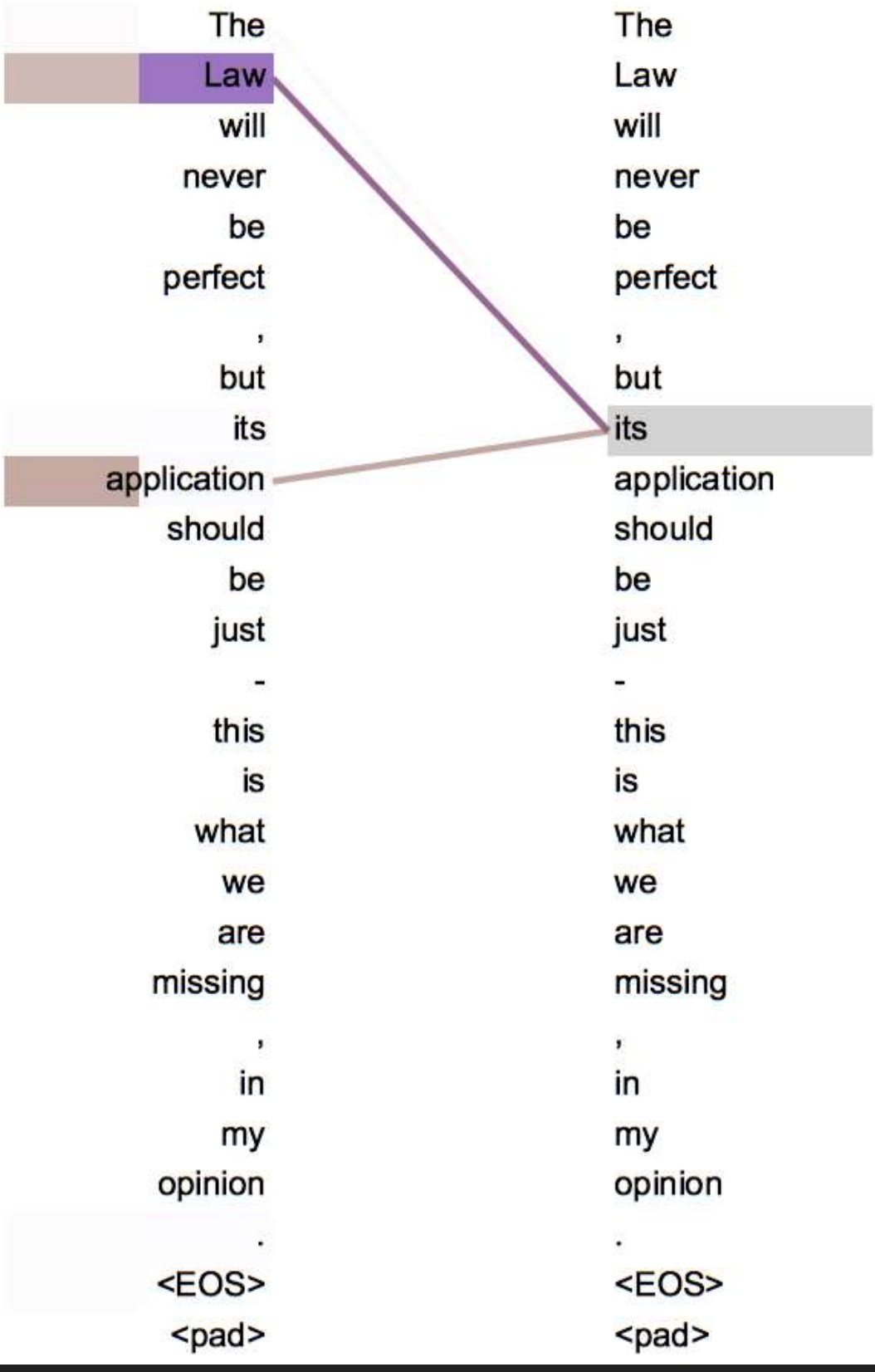
Results

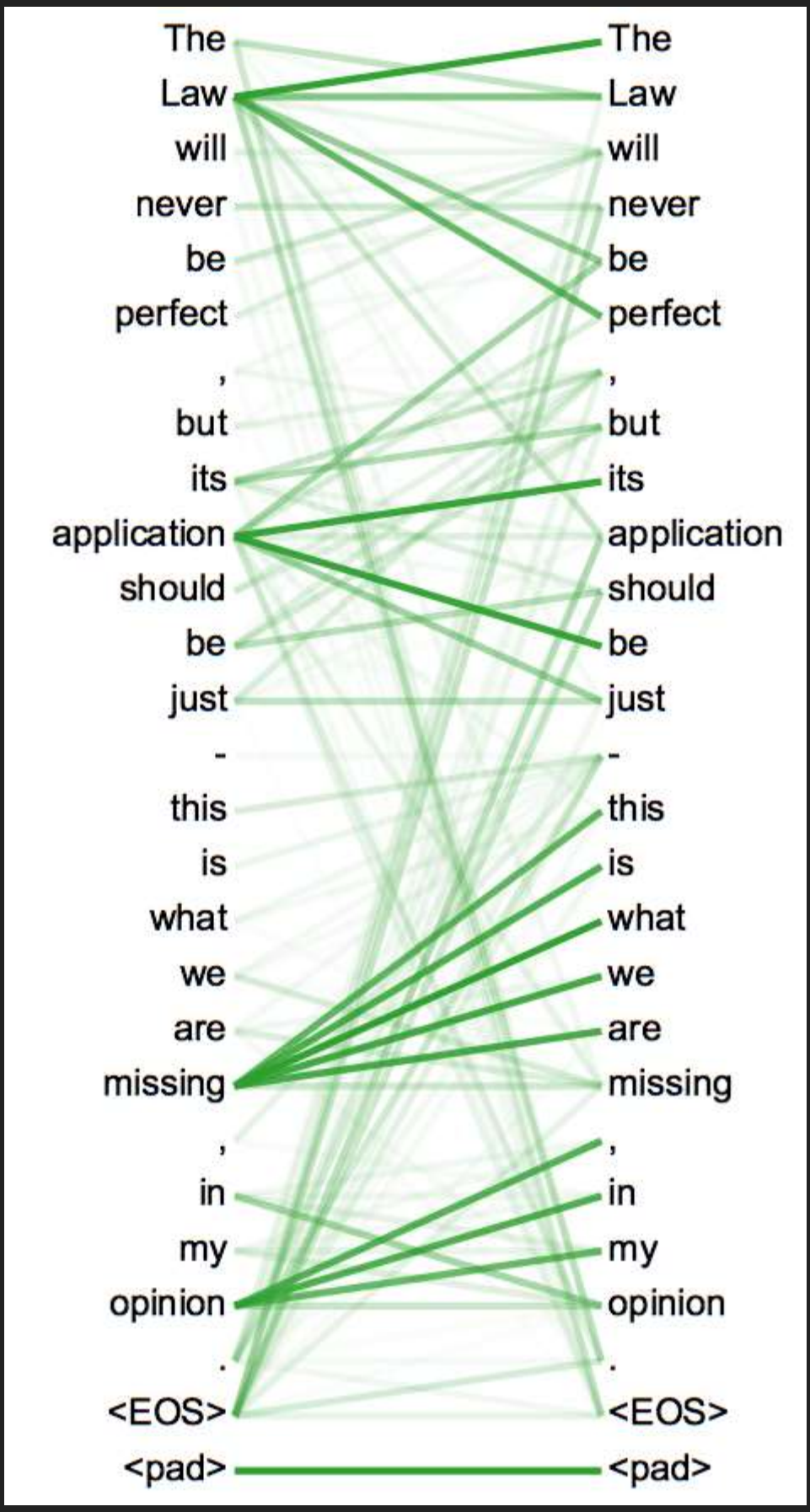
Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

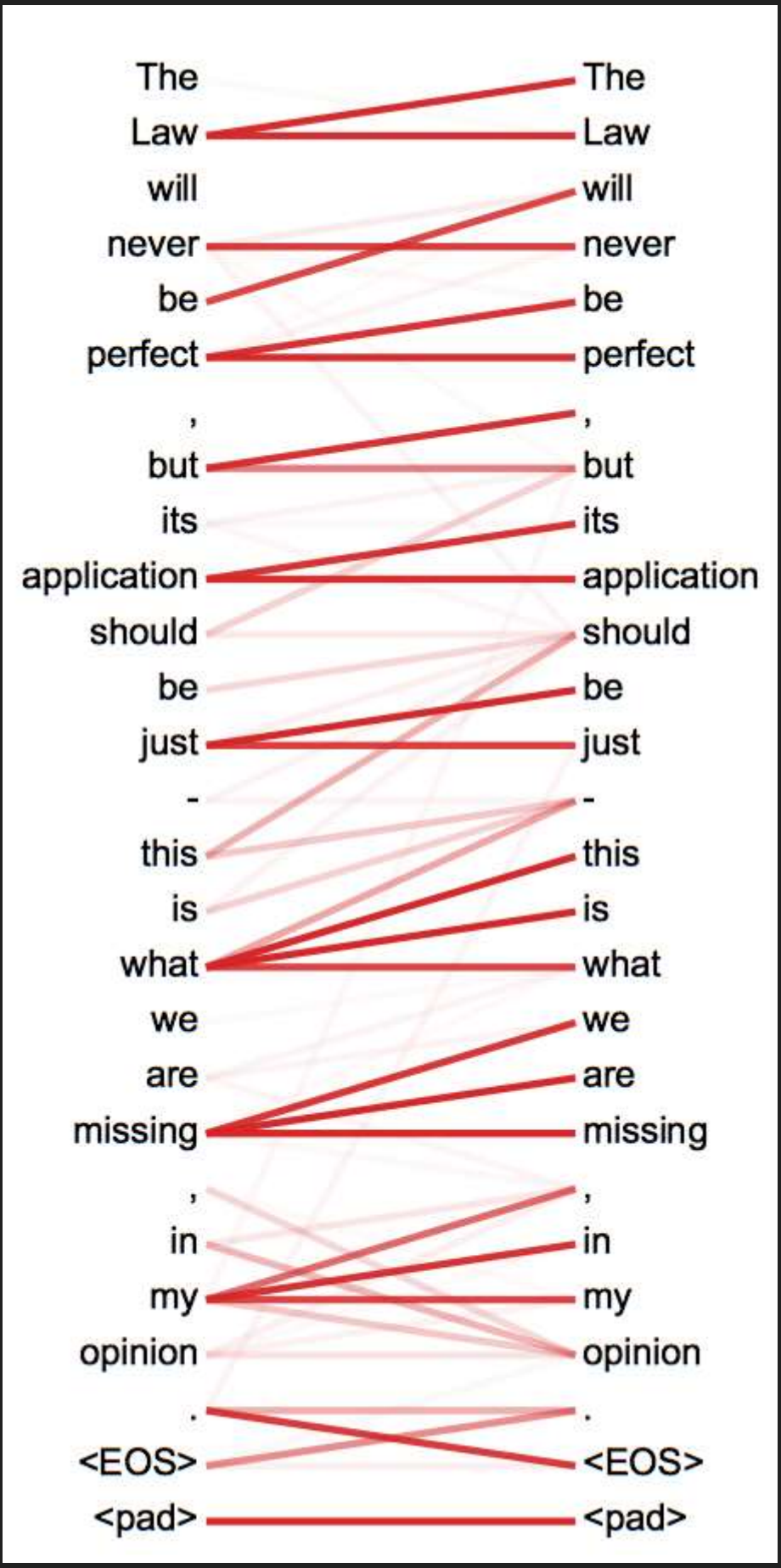
Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [17]	23.75			
Deep-Att + PosUnk [37]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [36]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [31]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [37]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [36]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	ϵ_{ls}	train steps	PPL (dev)	BLEU (dev)	params $\times 10^6$
base	6	512	2048	8	64	64	0.1	0.1	100K	4.92	25.8	65
(A)				1	512	512				5.29	24.9	
				4	128	128				5.00	25.5	
				16	32	32				4.91	25.8	
				32	16	16				5.01	25.4	
(B)					16					5.16	25.1	58
					32					5.01	25.4	60
(C)	2									6.11	23.7	36
	4									5.19	25.3	50
	8									4.88	25.5	80
		256			32	32				5.75	24.5	28
		1024			128	128				4.66	26.0	168
			1024							5.12	25.4	53
			4096							4.75	26.2	90
(D)							0.0			5.77	24.6	
							0.2			4.95	25.5	
								0.0		4.67	25.3	
								0.2		5.47	25.7	
(E)	positional embedding instead of sinusoids									4.92	25.7	
big	6	1024	4096	16			0.3		300K	**4.33**	**26.4**	213


It
is
in
this
spirit
that
a
majority
of
American
governments
have
passed
new
laws
since
2009
making
the
registration
or
voting
process
more
difficult
.
<EOS>
<pad>
<pad>
<pad>
<pad>
<pad>
<pad>
It
is
in
this
spirit
that
a
majority
of
American
governments
have
passed
new
laws
since
2009
making
the
registration
or
voting
process
more
difficult
.
<EOS>
<pad>
<pad>
<pad>
<pad>
<pad>
<pad>
<pad>


The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>
The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>


The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>
The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>


The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>
The
Law
will
never
be
perfect
,
but
its
application
should
be
just
-
this
is
what
we
are
missing
,
in
my
opinion
.
<EOS>
<pad>

Constituency Parsing

Parser	Training	WSJ 23 F1
Vinyals & Kaiser el al. (2014) [35]	WSJ only, discriminative	88.3
Petrov et al. (2006) [28]	WSJ only, discriminative	90.4
Zhu et al. (2013) [38]	WSJ only, discriminative	90.4
Dyer et al. (2016) [8]	WSJ only, discriminative	91.7
Transformer (4 layers)	WSJ only, discriminative	91.3
Zhu et al. (2013) [38]	semi-supervised	91.3
Huang & Harper (2009) [14]	semi-supervised	91.3
McClosky et al. (2006) [25]	semi-supervised	92.1
Vinyals & Kaiser el al. (2014) [35]	semi-supervised	92.1
Transformer (4 layers)	semi-supervised	92.7
Dyer et al. (2016) [8]	generative	93.3

Illia Polosukhin
NEAR.AI
@ilblackdragon, illia@near.ai

Questions?

Check out:
https://github.com/tensorflow/tensor2tensor
https://research.googleblog.com/2017/08/transformer-novel-neural-network.html
http://medium.com/@ilblackdragon