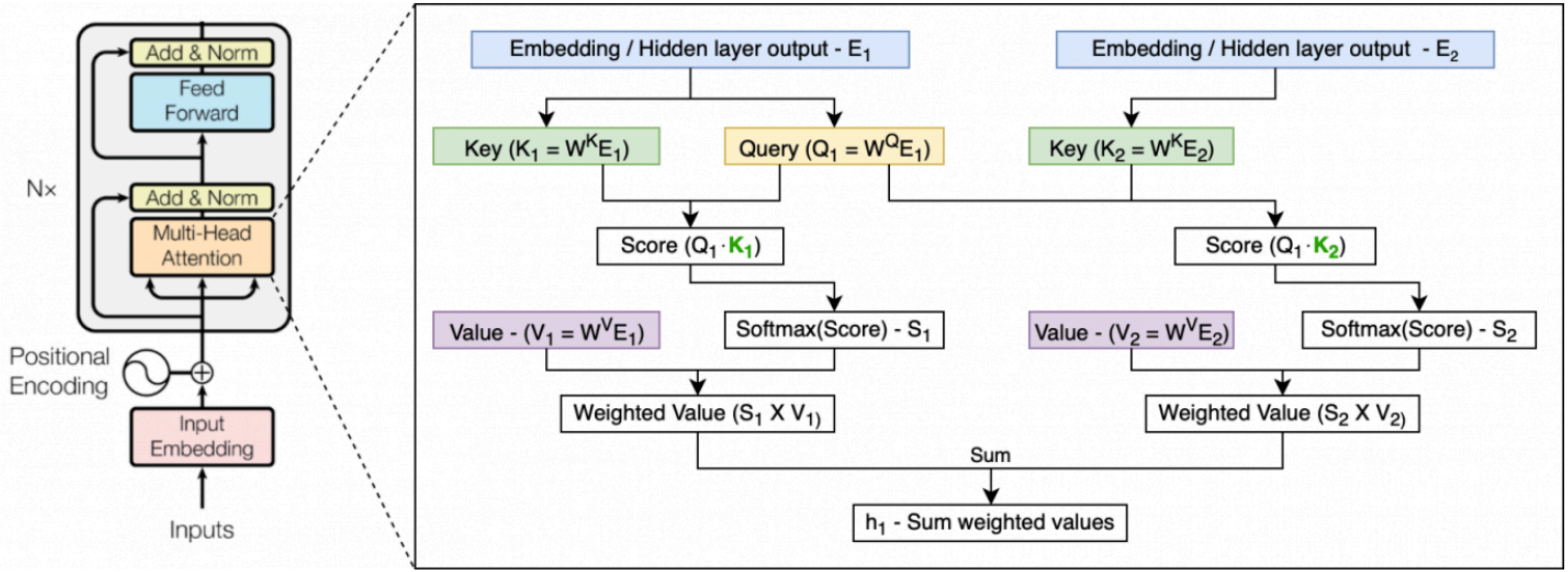
Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

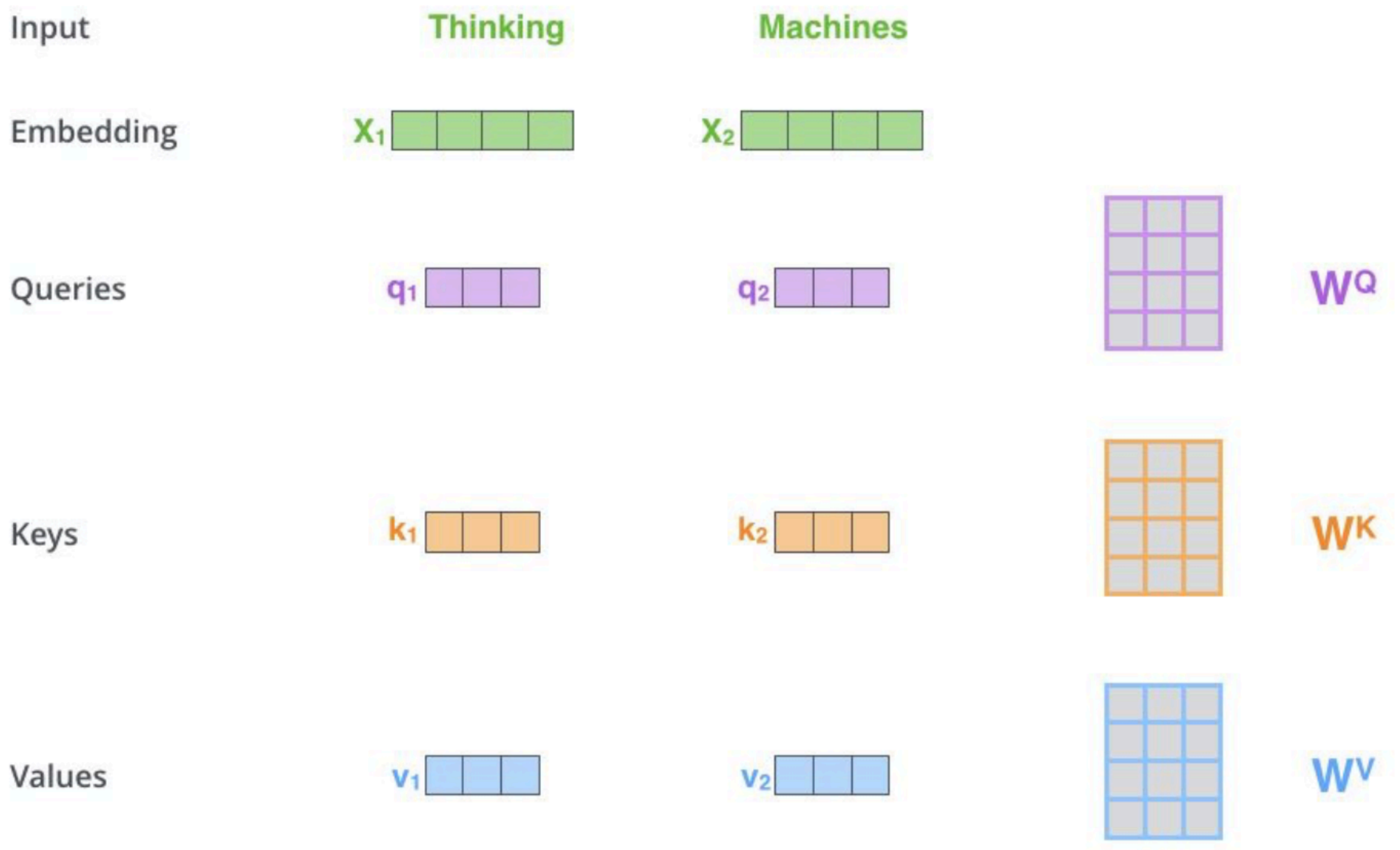
Zihang Dai*[12], Zhilin Yang*[12], Yiming Yang[1], Jaime Carbonell[1], Quoc V. Le[2], Ruslan Salakhutdinov[1]

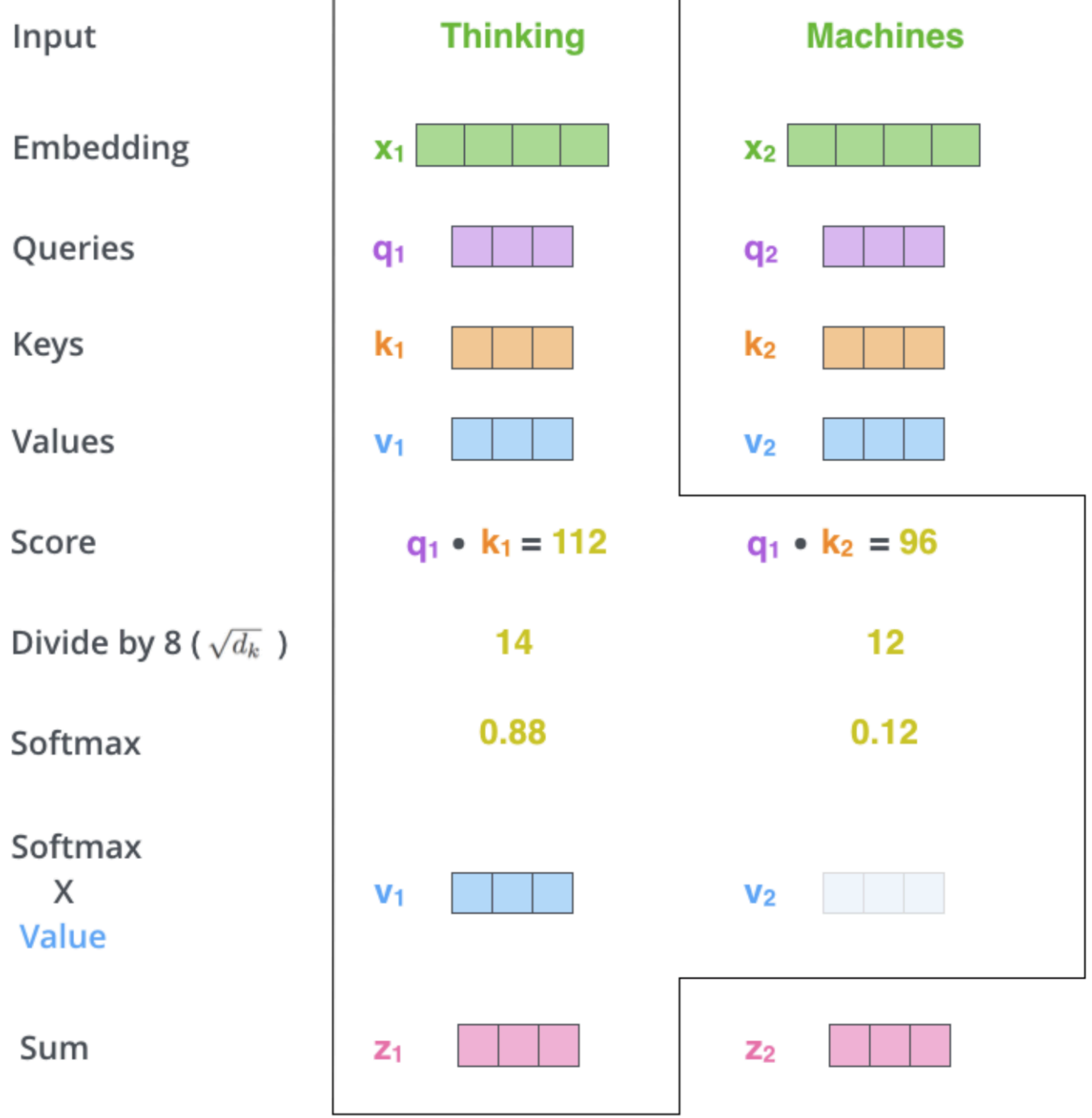
[1]Carnegie Mellon University, [2]Google Brain

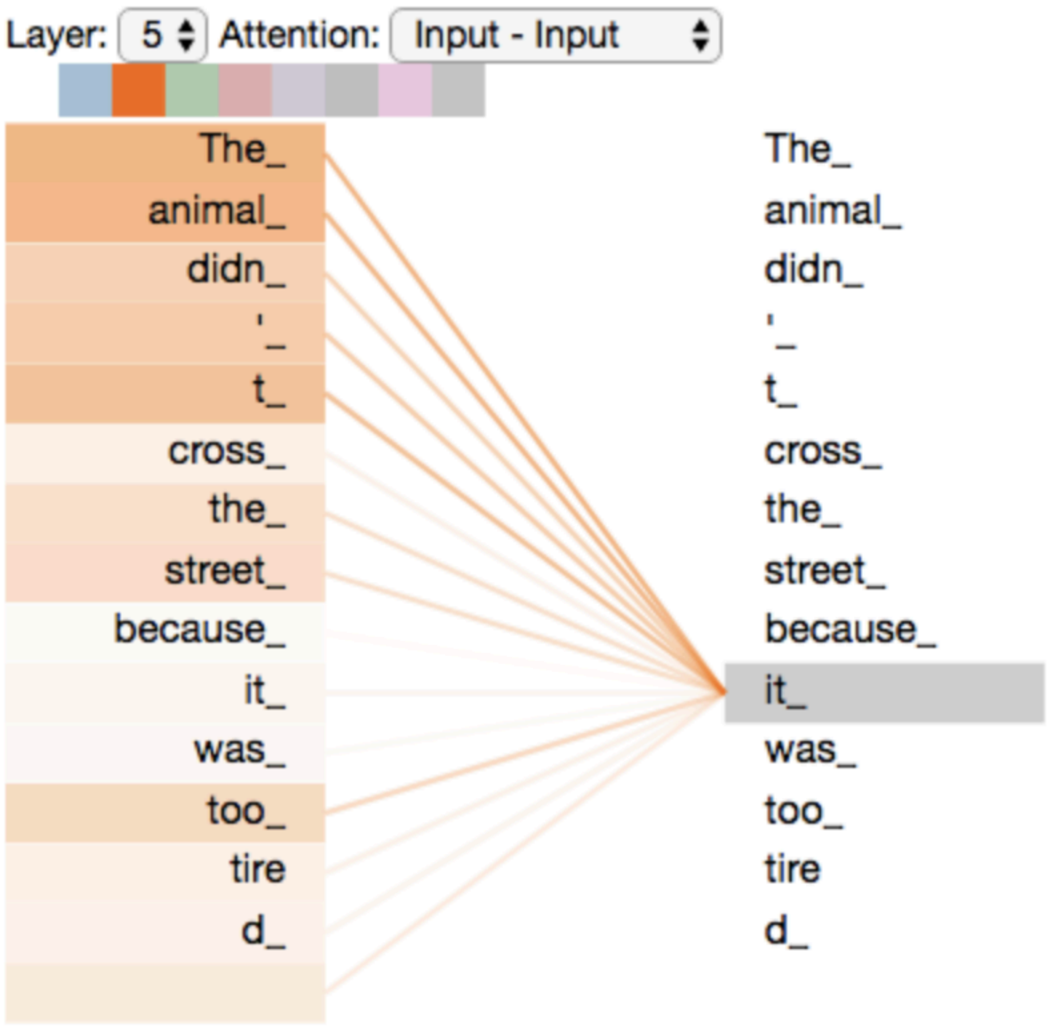
{dzihang,zhiliny,yiming,jgc,rsalakhu}@cs.cmu.edu, qvl@google.com

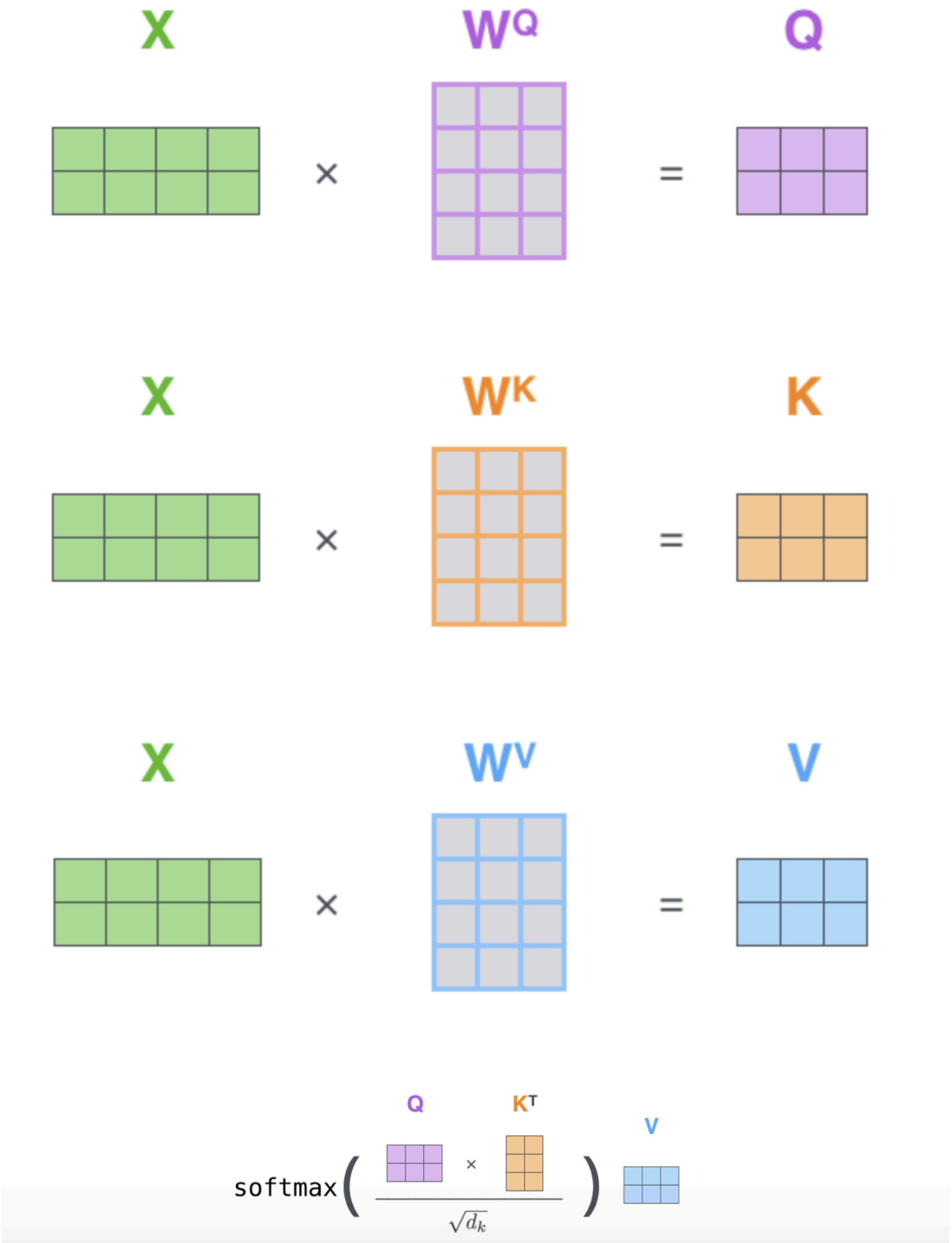
Review Transformer




Input
Thinking
Machines
Embedding
X1
X2
Queries
q1
q2
WQ
Keys
k1
k2
WK
Values
v1
v2
WV


Input
Thinking
Machines
Embedding
x1
x2
Queries
q1
q2
Keys
k1
k2
Values
v1
v2
Score
q1 • k1 = 112
q1 • k2 = 96
Divide by 8 ($\sqrt{d_k}$)
14
12
Softmax
0.88
0.12
Softmax
X
Value
v1
v2
Sum
z1
z2


Layer: 5 Attention: Input - Input
The_
animal_
didn_
'_
t_
cross_
the_
street_
because_
it_
was_
too_
tire
d_
The_
animal_
didn_
'_
t_
cross_
the_
street_
because_
it_
was_
too_
tire
d_

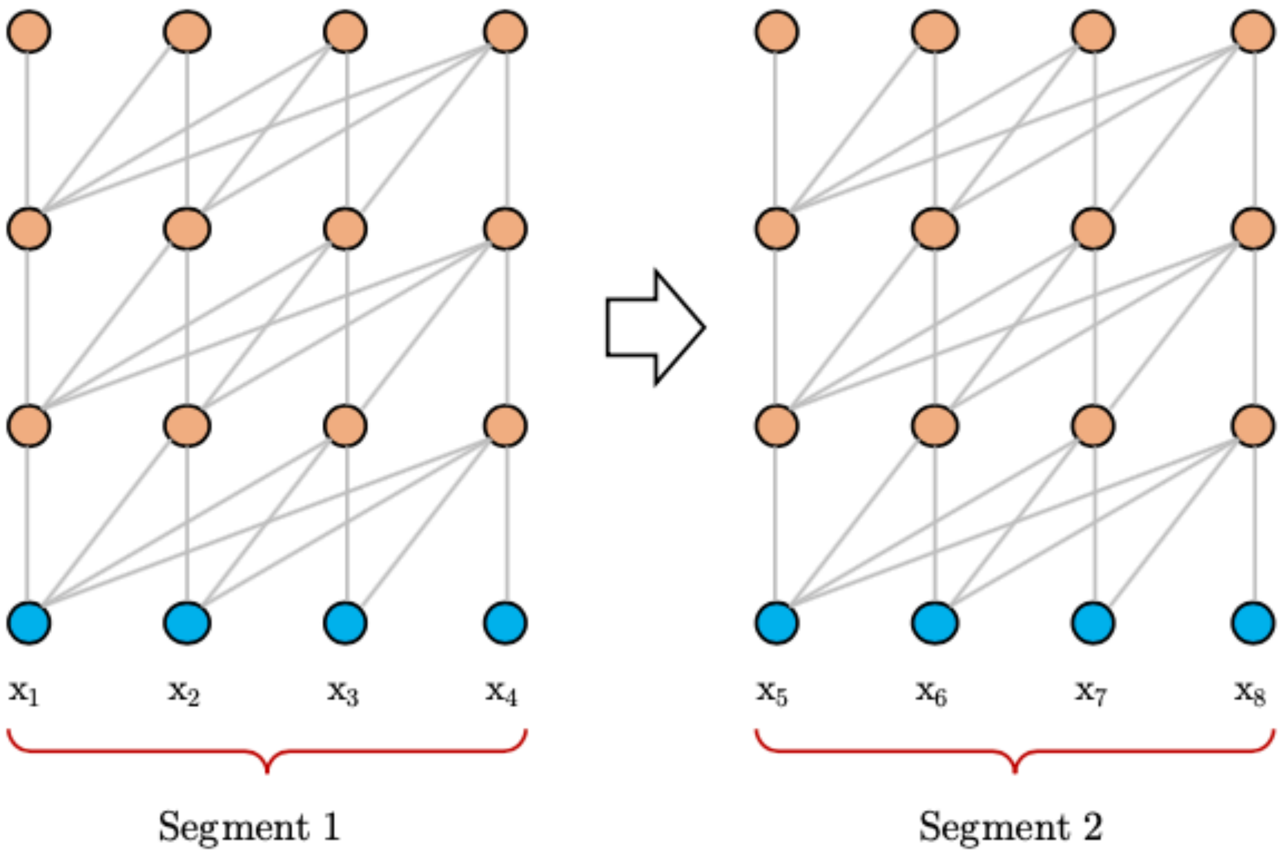

X
W^Q
Q
X
W^K
K
X
W^V
V
Q
K^T
V
softmax
√d_k

Vanilla Transformer

Problem: How to train a Transformer to effectively encode an arbitrarily long context into a fixed size representation

——Split the entire corpus into shorter segments

Vanilla training

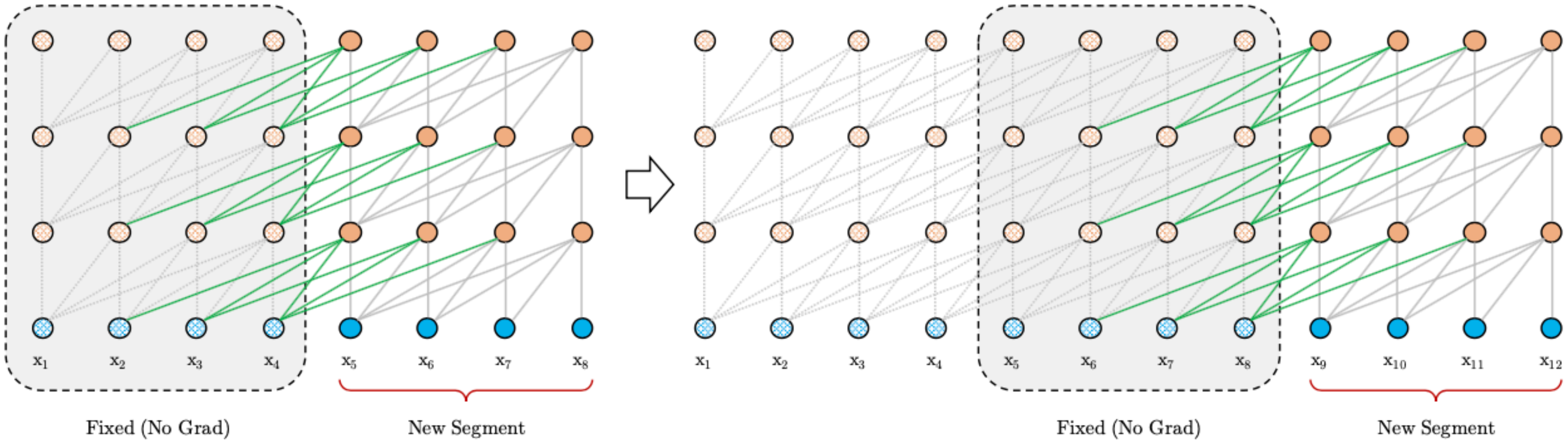


分割出来在语义上可能不完整

Vanilla training

- Information never flows across segments
- largest possible dependency is bound by sequence length
- longer sentences/sequences get split up causing context fragmentation

XL training——Segment Level Recurrence



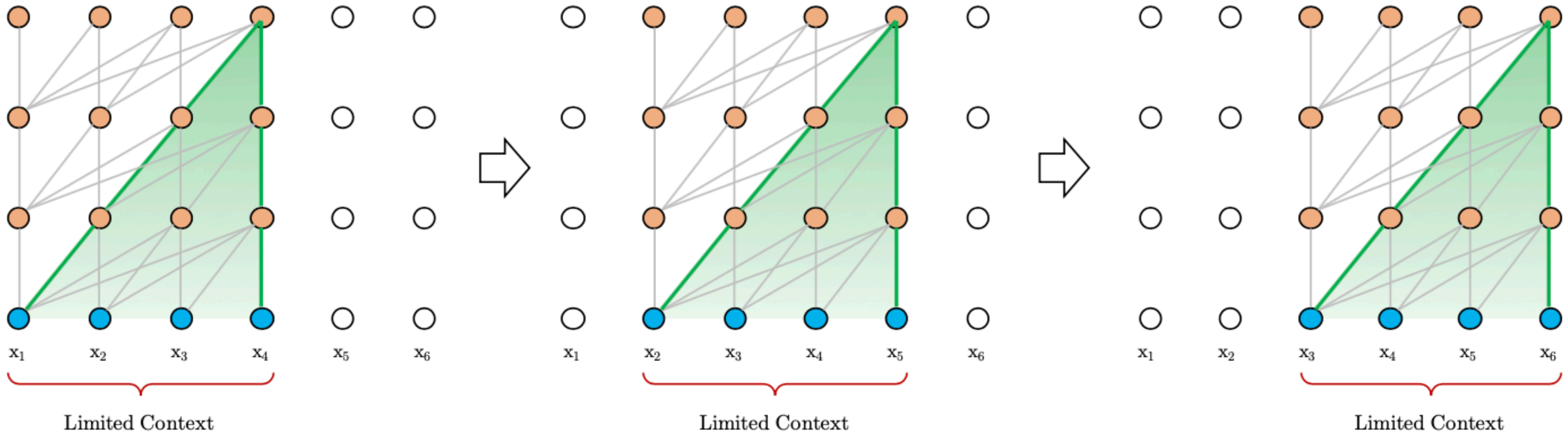
In formula

$$\widetilde{\mathbf{h}}_{\tau+1}^{n-1} = \left[\mathrm{SG}(\mathbf{h}_{\tau}^{n-1}) \circ \mathbf{h}_{\tau+1}^{n-1}\right],$$

$$\mathbf{q}_{\tau+1}^{n}, \mathbf{k}_{\tau+1}^{n}, \mathbf{v}_{\tau+1}^{n} = \mathbf{h}_{\tau+1}^{n-1}\mathbf{W}_{q}^{\top}, \widetilde{\mathbf{h}}_{\tau+1}^{n-1}\mathbf{W}_{k}^{\top}, \widetilde{\mathbf{h}}_{\tau+1}^{n-1}\mathbf{W}_{v}^{\top}$$

$$\mathbf{h}_{\tau+1}^{n} = \text{Transformer-Layer}\left(\mathbf{q}_{\tau+1}^{n}, \mathbf{k}_{\tau+1}^{n}, \mathbf{v}_{\tau+1}^{n}\right).$$

Vanilla Prediction



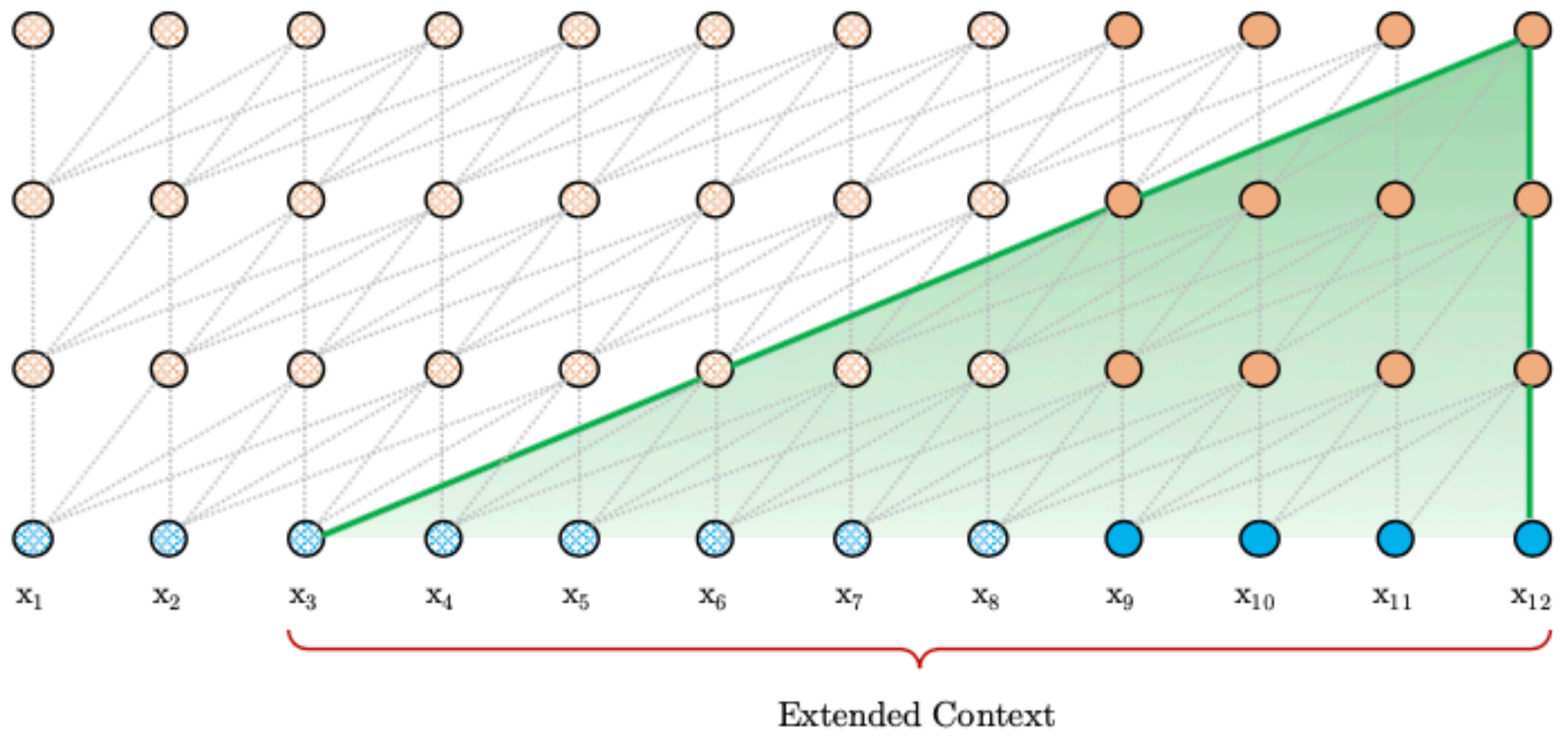
Vanilla Prediction

- Consumes one segment at a time, but only does one prediction
- Relieves context fragmentation
- Extremely expensive

碎片问题

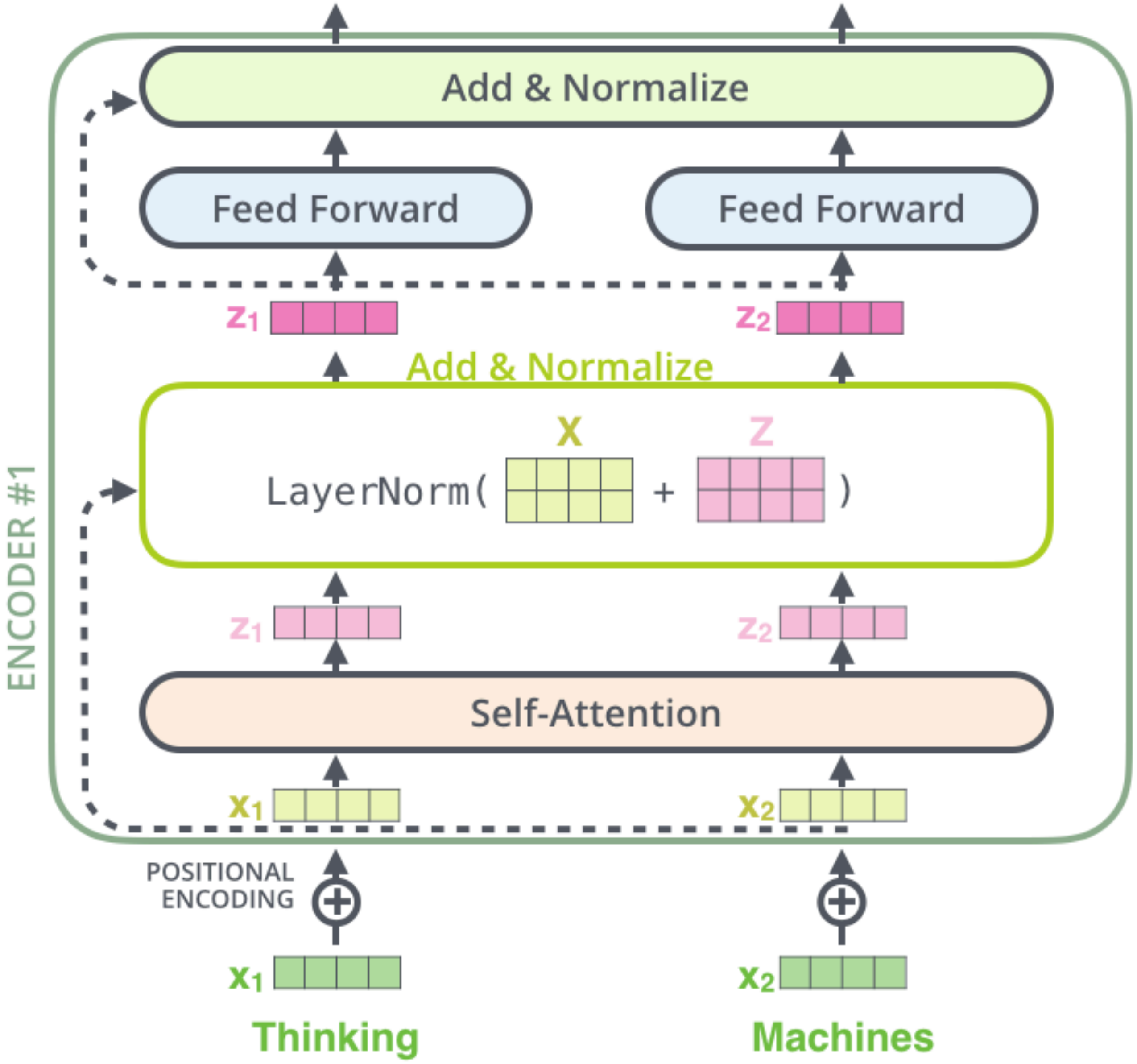
XL prediction

训练的时候，缓存一个segment
预测的时候，一次缓存多个segment



Position wise embeddings

Vanilla Embeddings



Vanilla Embeddings--Position embedding

$\mathbf{E}_{\mathbf{s}_\tau} \in \mathbb{R}^{L\times d}$ Word embeddings

$\mathbf{U}_{1:L}$ Position wise embeddings

$$\mathbf{h}_\tau = f(\mathbf{h}_{\tau-1}, \mathbf{E}_{\mathbf{s}_\tau} + \mathbf{U}_{1:L})$$

$$\mathbf{h}_{\tau+1} = f(\mathbf{h}_\tau, \mathbf{E}_{\mathbf{s}_{\tau+1}} + \mathbf{U}_{1:L})$$

What to change?

Relative position embeddings

- Vanilla transformer

$$\mathbf{A}_{i,j}^{\text{abs}} = \underbrace{\mathbf{E}_{x_i}^\top \mathbf{W}_q^\top \mathbf{W}_k \mathbf{E}_{x_j}}_{(a)} + \underbrace{\mathbf{E}_{x_i}^\top \mathbf{W}_q^\top \mathbf{W}_k \mathbf{U}_j}_{(b)} + \underbrace{\mathbf{U}_i^\top \mathbf{W}_q^\top \mathbf{W}_k \mathbf{E}_{x_j}}_{(c)} + \underbrace{\mathbf{U}_i^\top \mathbf{W}_q^\top \mathbf{W}_k \mathbf{U}_j}_{(d)}.$$

- Transformer-XL

$$\mathbf{A}_{i,j}^{\text{rel}} = \underbrace{\mathbf{E}_{x_i}^\top \mathbf{W}_q^\top \mathbf{W}_{k,E} \mathbf{E}_{x_j}}_{(a)} + \underbrace{\mathbf{E}_{x_i}^\top \mathbf{W}_q^\top \mathbf{W}_{k,R} \mathbf{R}_{i-j}}_{(b)} + \underbrace{u^\top \mathbf{W}_{k,E} \mathbf{E}_{x_j}}_{(c)} + \underbrace{v^\top \mathbf{W}_{k,R} \mathbf{R}_{i-j}}_{(d)}.$$

Original self attention in other words

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Extend Self attention

- Extend Qi*Ki by four terms:
  - Content Weight: the original score without the addition of the original positional encoding of course
  - Positional bias with respect to the current content (Qi). Sinusoidal function that receives the distance between tokens (e.g. i-j).
  - A learned global content bias - vector for adjusting key importance
  - A learned global position bias

Transformer-XL

$$\mathbf{h}_\tau^0 := \mathbf{E}_{\mathbf{s}_\tau}$$

For $n = 1, \ldots, N$:

$$\begin{aligned}
\widetilde{\mathbf{h}}_\tau^{n-1} &= \left[\mathrm{SG}(\mathbf{m}_\tau^{n-1}) \circ \mathbf{h}_\tau^{n-1}\right] \\
\mathbf{q}_\tau^n, \mathbf{k}_\tau^n, \mathbf{v}_\tau^n &= \mathbf{h}_\tau^{n-1} {\mathbf{W}_q^n}^\top, \widetilde{\mathbf{h}}_\tau^{n-1} {\mathbf{W}_{k,E}^n}^\top, \widetilde{\mathbf{h}}_\tau^{n-1} {\mathbf{W}_v^n}^\top \\
\mathbf{A}_{\tau,i,j}^n &= {\mathbf{q}_{\tau,i}^n}^\top \mathbf{k}_{\tau,j}^n + {\mathbf{q}_{\tau,i}^n}^\top \mathbf{W}_{k,R}^n \mathbf{R}_{i-j} + u^\top \mathbf{k}_{\tau,j} + v^\top \mathbf{W}_{k,R}^n \mathbf{R}_{i-j} \\
\mathbf{a}_\tau^n &= \text{Masked-Softmax}(\mathbf{A}_\tau^n)\mathbf{v}_\tau^n \\
\mathbf{o}_\tau^n &= \text{LayerNorm}(\text{Linear}(\mathbf{a}_\tau^n) + \mathbf{h}_\tau^{n-1}) \\
\mathbf{h}_\tau^n &= \text{Positionwise-Feed-Forward}(\mathbf{o}_\tau^n)
\end{aligned}$$

$W^n_{k,E}$ Content based

R_{i-j} Relative position embedding

v^T Global content bias

$W^n_{k,R}$ Location based

u^T Global position bias

WikiText-103

Model	#Params	Validation PPL	Test PPL
Grave et al. (2016b) – LSTM	-	-	48.7
Bai et al. (2018) – TCN	-	-	45.2
Dauphin et al. (2016) – GCNN-8	-	-	44.9
Grave et al. (2016b) – LSTM + Neural cache	-	-	40.8
Dauphin et al. (2016) – GCNN-14	-	-	37.2
Merity et al. (2018) – 4-layer QRNN	151M	32.0	33.0
Rae et al. (2018) – LSTM + Hebbian + Cache	-	29.7	29.9
Ours – Transformer-XL Standard	151M	**23.1**	**24.0**
Baevski & Auli (2018) – adaptive input$^{\diamond}$	247M	19.8	20.5
Ours – Transformer-XL Large	257M	**17.7**	**18.3**

Attention Length 384 training, 1600 test

103M training tokens from 28k articles, with 3.6k tokens per article

enwiki8

Model	#Params	Test bpc
Ha et al. (2016) – LN HyperNetworks	27M	1.34
Chung et al. (2016) – LN HM-LSTM	35M	1.32
Zilly et al. (2016) – Recurrent highway networks	46M	1.27
Mujika et al. (2017) – Large FS-LSTM-4	47M	1.25
Krause et al. (2016) – Large mLSTM	46M	1.24
Knol (2017) – cmix v13	-	1.23
Al-Rfou et al. (2018) – 12-layer Transformer	44M	1.11
Ours – 12-layer Transformer-XL	41M	**1.06**
Al-Rfou et al. (2018) – 64-layer Transformer	235M	1.06
Ours – 18-layer Transformer-XL	88M	1.03
Ours – 24-layer Transformer-XL	277M	**0.99**

Attention Length 784 training, 3200 test

text8

Model	#Params	Test bpc
Cooijmans et al. (2016) – BN-LSTM	-	1.36
Chung et al. (2016) – LN HM-LSTM	35M	1.29
Zilly et al. (2016) – Recurrent highway networks	45M	1.27
Krause et al. (2016) – Large mLSTM	45M	1.27
Al-Rfou et al. (2018) – 12-layer Transformer	44M	1.18
Al-Rfou et al. (2018) – 64-layer Transformer	235M	1.13
Ours – 24-layer Transformer-XL	277M	**1.08**

Attention Length 784 training, 3200 test

Summary

- Enable language modeling with self-attention architecture beyond a fixed length context. (Recurrence in purely self-attentive model)
- can learn longer dependency
  - 80% and 450% more than RNN and vanilla transformer
- 1,800 times faster than vanilla transformer