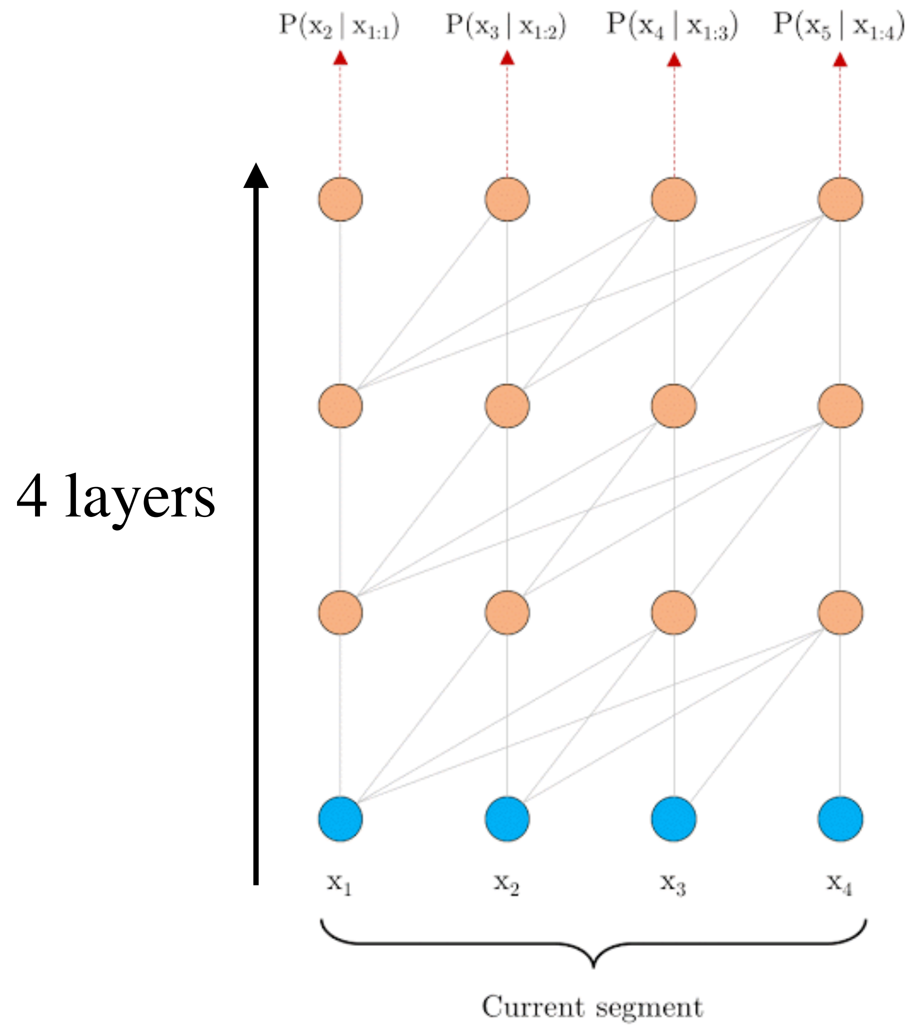
Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

Zihang Dai*[1,2], Zhilin Yang*[1,2], 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

WenWei

Vanilla Transformer



$$\begin{cases} X = \{x_1, x_2, \cdots, x_n\},\ if\ n \le 512 \\ X_1 = \{x_1^1, \cdots, x_{512}^1\}, X_2 = \{x_1^2, \cdots, x_{n-512}^2\},\ if\ n\ \ge 512 \end{cases}$$

- Context fragmentation
- Absolute positional encoding

Vanilla Transformer

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \qquad \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,m} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,m} \end{bmatrix}$$

$$\begin{aligned} A^{abs}_{i,j} = q_i^T k_j &= (W_q(E_i + U_i))^T(W_k(E_j + U_j)) \\ &= (E_i + U_i)^T W_q^T W_k (E_j + U_j) \\ &= \underset{(a)}{\underline{E_i^T W_q^T W_k E_j}} + \underset{(b)}{\underline{E_i^T W_q^T W_k U_j}} + \underset{(c)}{\underline{U_i^T W_q^T W_k E_j}} + \underset{(d)}{\underline{U_i^T W_q^T W_k U_j}} \end{aligned}$$

- E_i : i^{th} *word embedding*, E_j : j^{th} *word embedding*
- U_i : i^{th} *positional encoding*, U_i : i^{th} *positional encoding*

- (a): *dot product of* i^{th} *word embedding and* j^{th} *word embedding*
- (b): *dot product of* i^{th} *word embedding and* j^{th} *positional encoding*
- (c): *dot product of* i^{th} *positional encoding and* j^{th} *word embedding*
- (d): *dot product of* i^{th} *positional encoding and* j^{th} *positional encoding*

Vanilla Transformer

$$
\begin{aligned}
A_{i,j}^{abs} = q_i^T k_j &= (W_q(E_i + U_i))^T (W_k(E_j + U_j)) \\
&= (E_i + U_i)^T W_q^T W_k (E_j + U_j) \\
&= \underbrace{E_i^T W_q^T W_k E_j}_{(a)} + \underbrace{E_i^T W_q^T W_k U_j}_{(b)} + \underbrace{U_i^T W_q^T W_k E_j}_{(c)} + \underbrace{U_i^T W_q^T W_k U_j}_{(d)}
\end{aligned}
$$

$(b)(c)(d)$ Absolute positional encoding:

Using relative positional encodings rather than absolute ones, in order to enable state reuse without causing temporal confusion.

τ^{th} *segment* : $\qquad\qquad (\tau+1)^{th}$ *segment* :

$$
\left[\begin{array}{ccc|ccc}
q_0 U_0 & \cdots & q_0 U_{M-1} & q_0 U_0 & \cdots & q_0 U_{L-1} \\
q_1 U_0 & \cdots & q_1 U_{M-1} & q_1 U_0 & \cdots & q_1 U_{L-1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
q_{L-1} U_0 & \cdots & q_{L-1} U_{M-1} & q_{L-1} U_0 & \cdots & q_{L-1} U_{L-1}
\end{array}\right]
$$

$h_{\tau+1} = f(h_\tau, E_{s_{\tau+1}} + U_{1:L})$

$h_\tau = f(h_{\tau-1}, E_{s_\tau} + U_{1:L})$

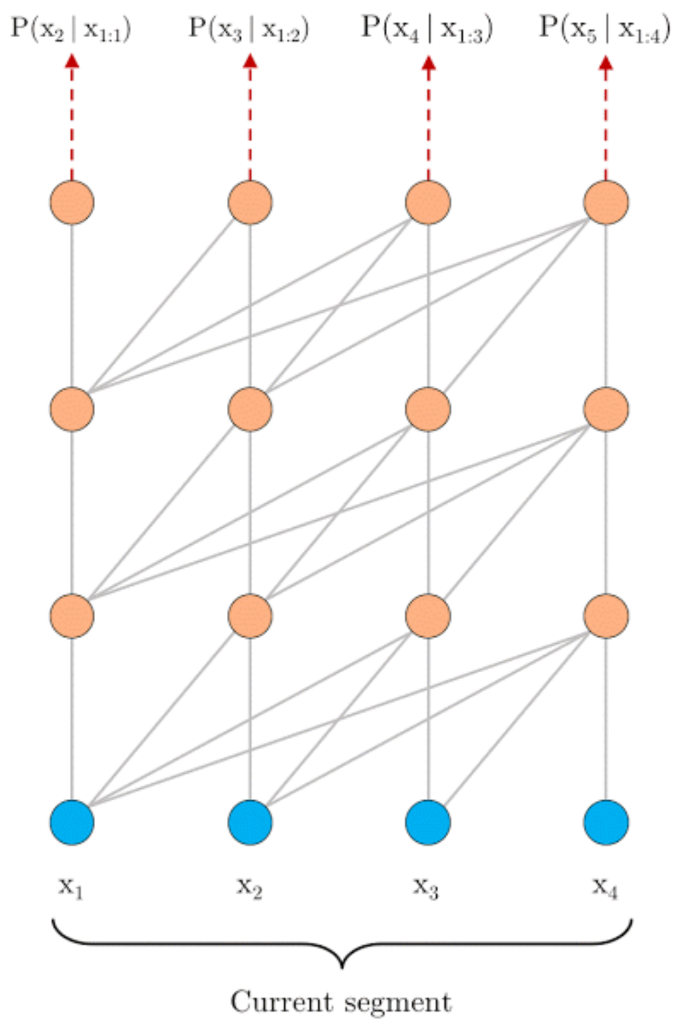
τ : τ^{th} *segment*

E_{s_τ} : *word embedding of* s_τ

$U_{1:L}$: *positional encoding*

Model has no information to distinguish the positional difference between s_τ and $s_{\tau+1}$.

Transformer-XL



Reuse previous (τ^{th}) hidden state:

$$\tilde{h}_{\tau+1}^{n-1} = [SG(h_{\tau}^{n+1}) \circ h_{\tau+1}^{n-1}],\ SG() : stop\ gradient$$

$$q_{\tau+1}^{n}, k_{\tau+1}^{n}, v_{\tau+1}^{n} = h_{\tau+1}^{n-1} W_q^T,\ \tilde{h}_{\tau+1}^{n-1} W_k^T,\ \tilde{h}_{\tau+1}^{n-1} W_v^T$$

$$h_{\tau+1}^{n} = Transformer - Layer(q_{\tau+1}^{n}, k_{\tau+1}^{n}, v_{\tau+1}^{n})$$

Transformer-XL

Vanilla Transformer:

$$
\begin{aligned}
A_{i,j}^{abs} = q_i^T k_j &= (W_q(E_i + U_i))^T (W_k(E_j + U_j)) \\
&= (E_i + U_i)^T W_q^T W_k (E_j + U_j) \\
&= \underbrace{E_i^T W_q^T W_k E_j}_{(a)} + \underbrace{E_i^T W_q^T W_k U_j}_{(b)} + \underbrace{U_i^T W_q^T W_k E_j}_{(c)} + \underbrace{U_i^T W_q^T W_k U_j}_{(d)}
\end{aligned}
$$

Transformer-XL:

$$
A_{i,j}^{rel} = E_i^T W_q^T W_{k,E} E_j + E_i^T W_q^T W_{k,R} R_{i-j} + u^T W_{k,E} E_j + v^T W_{k,R} R_{i-j}
$$

: Separate the two weight matrices $W_{k,E}$ and $W_{k,E}$ for producing the content-based key vectors and location-based key vectors respectively.

Transformer: $K = W_k \left(\boxed{E_j} + \boxed{U_j} \right)$

Transformer-XL: $K = W_{k,E} \boxed{E_j} + W_{K,R} \boxed{R_{i-j}}$

Transformer-XL

Vanilla Transformer:

$$
\begin{aligned}
A_{i,j}^{abs} = q_i^T k_j &= (W_q(E_i + U_i))^T (W_k(E_j + U_j)) \\
&= (E_i + U_i)^T W_q^T W_k (E_j + U_j) \\
&= \underbrace{E_i^T W_q^T W_k E_j}_{(a)} + \underbrace{E_i^T W_q^T W_k U_j}_{(b)} + \underbrace{U_i^T W_q^T W_k E_j}_{(c)} + \underbrace{U_i^T W_q^T W_k U_j}_{(d)}
\end{aligned}
$$

Transformer-XL:

$$
A_{i,j}^{rel} = E_i^T W_q^T W_{k,E} E_j + E_i^T W_q^T W_{k,R} R_{i-j} + u^T W_{k,E} E_j + v^T W_{k,R} R_{i-j}
$$

: Replace absolute positional embedding U_j with relative positional embedding R_{i-j}.

(b)

τ^{th} *segment* : $\qquad$ $(\tau + 1)^{th}$ *segment* :

Transformer:

$$
\left[\begin{array}{ccc|ccc}
q_0 U_0 & \cdots & q_0 U_{M-1} & q_0 U_0 & \cdots & q_0 U_{L-1} \\
q_1 U_0 & \cdots & q_1 U_{M-1} & q_1 U_0 & \cdots & q_1 U_{L-1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
q_{L-1} U_0 & \cdots & q_{L-1} U_{M-1} & q_{L-1} U_0 & \cdots & q_{L-1} U_{L-1}
\end{array}\right]
$$

τ^{th} *segment* : $\qquad$ $(\tau + 1)^{th}$ *segment* :

Transformer-XL:

$$
\left[\begin{array}{ccc|ccc}
q_0 R_0 & \cdots & q_0 R_{-M+1} & q_0 R_{-M} & \cdots & q_0 R_{-M-L+1} \\
q_1 R_1 & \cdots & q_1 R_M & q_1 R_{-M+1} & \cdots & q_1 R_{L-1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
q_{L-1} R_{L-1} & \cdots & q_{L-1} R_{L-M} & q_{L-1} R_{L-M-1} & \cdots & q_{L-1} R_{-M}
\end{array}\right]
$$

Transformer-XL

Vanilla Transformer:

$$
\begin{aligned}
A_{i,j}^{abs} = q_i^T k_j &= (W_q(E_i + U_i))^T (W_k(E_j + U_j)) \\
&= (E_i + U_i)^T W_q^T W_k (E_j + U_j) \\
&= \underbrace{E_i^T W_q^T W_k E_j}_{(a)} + \underbrace{E_i^T W_q^T W_k U_j}_{(b)} + \underbrace{U_i^T W_q^T W_k E_j}_{(c)} + \underbrace{U_i^T W_q^T W_k U_j}_{(d)}
\end{aligned}
$$

Transformer-XL:

$$
A_{i,j}^{rel} = E_i^T W_q^T W_{k,E} E_j + E_i^T W_q^T W_{k,R} R_{i-j} + u^T W_{k,E} E_j + v^T W_{k,R} R_{i-j}
$$

: The query vector is the same for all query positions, and the attentive bias should remain the same regardless of the query positions.

Transformer-XL:

$$A_{i,j}^{rel} = \underbrace{E_i^T W_q^T W_{k,E} E_j}_{(a)} + \underbrace{E_i^T W_q^T W_{k,R} R_{i-j}}_{(b)} + \underbrace{u^T W_{k,E} E_j}_{(c)} + \underbrace{v^T W_{k,R} R_{i-j}}_{(d)}$$

- (a) represents content-based addressing.
- (b) captures a content-dependent positional bias.
- (c) governs a global content bias.
- (d) encodes a global positional bias.

Transformer-XL architecture:

$$\tilde{h}_\tau^{n-1} = [SG(h_{\tau-1}^{n-1}) \circ h_\tau^{n-1}]$$

$$q_\tau^n, k_\tau^n, v_\tau^n = h_\tau^{n-1} W_q^{n\top}, \tilde{h}_\tau^{n-1} W_{k,E}^{n}{}^\top, \tilde{h}_\tau^{n-1} W_v^{n\top}$$

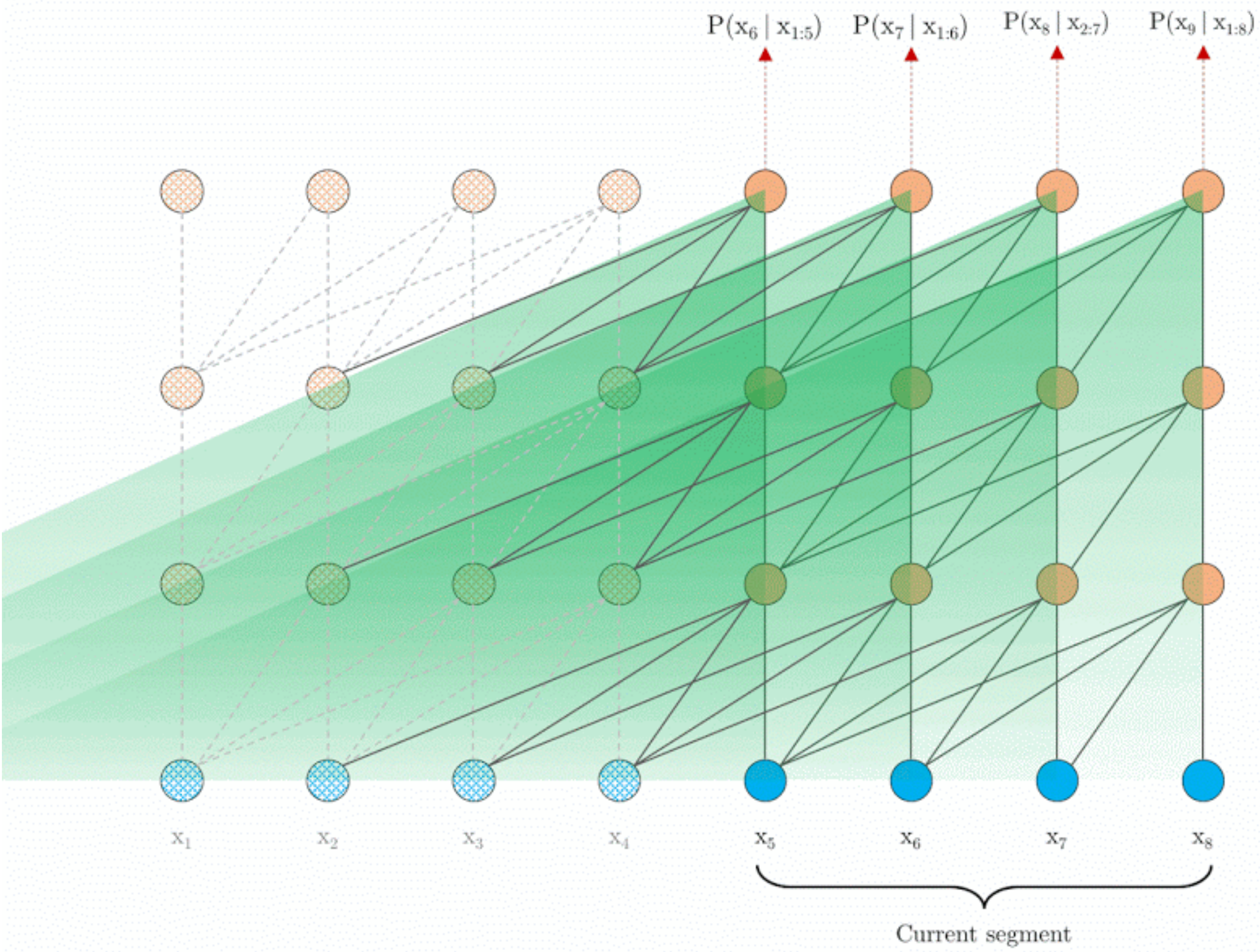
$$A_{\tau,i,j}^n = q_{\tau,i}^n{}^\top k_{\tau,j}^n + q_{\tau,i}^n{}^\top W_{k,R}^n R_{i-j} + u^\top k_{\tau,j} + v^\top W_{k,R}^n R_{i-j}$$

$$a_\tau^n = Mask - Softmax(A_\tau^n) v_\tau^n$$

$$o_\tau^n = LayNrom(Linear(a_\tau^n) + h_\tau^{n-1})$$

$$h_\tau^n = Positionwise - Feed - Forward(o_\tau^n)$$

Transformer-XL



Evaluation

Model	#Param	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) - QRNN	151M	33.0
Rae et al. (2018) - Hebbian + Cache	-	29.9
Ours - Transformer-XL Standard	151M	**24.0**
Baevski and Auli (2018) - Adaptive Input $^{\diamond}$	247M	20.5
Ours - Transformer-XL Large	257M	**18.3**

Table 1: Comparison with state-of-the-art results on WikiText-103. $^{\diamond}$ indicates contemporary work.

Model	#Param	bpc
Ha et al. (2016) - LN HyperNetworks	27M	1.34
Chung et al. (2016) - LN HM-LSTM	35M	1.32
Zilly et al. (2016) - RHN	46M	1.27
Mujika et al. (2017) - 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) - 12L Transformer	44M	1.11
Ours - 12L Transformer-XL	41M	**1.06**
Al-Rfou et al. (2018) - 64L Transformer	235M	1.06
Ours - 18L Transformer-XL	88M	1.03
Ours - 24L Transformer-XL	277M	**0.99**

Table 2: Comparison with state-of-the-art results on enwik8.

- WikiText-103: Word-level dataset with an average length of 3.6K tokens per article.
- enwik8 contains 100M bytes of un- processed Wikipedia text.

Evaluation

Remark	Recurrence	Encoding	Loss	PPL init	PPL best	Attn Len
Transformer-XL (128M)	✓	Ours	Full	**27.02**	**26.77**	**500**
-	✓	Shaw et al. (2018)	Full	27.94	27.94	256
-	✓	Ours	Half	28.69	28.33	460
-	✗	Ours	Full	29.59	29.02	260
-	✗	Ours	Half	30.10	30.10	120
-	✗	Shaw et al. (2018)	Full	29.75	29.75	120
-	✗	Shaw et al. (2018)	Half	30.50	30.50	120
-	✗	Vaswani et al. (2017)	Half	30.97	30.97	120
Transformer (128M)†	✗	Al-Rfou et al. (2018)	Half	31.16	31.16	120
					23.09	**640**
Transformer-XL (151M)	✓	Ours	Full	23.43	23.16	450
					23.35	300

Table 6: Ablation study on WikiText-103. For the first two blocks, we use a slightly smaller model (128M parameters). † indicates that the corresponding row is reduced to the same setting as the Transformer network in (Al-Rfou et al., 2018), except that two auxiliary losses are not implemented in our experiments. "PPL init" refers to using the same length as training. "PPL best" indicates the perplexity obtained by using the optimal length. "Attn Len" is the shortest possible attention length during evaluation to achieve the corresponding result (PPL best). Increasing the attention length during evaluation improves performance only when our positional encoding is used. The "Transformer-XL (151M)" setting uses a standard parameter budget as previous work (Merity et al., 2018), where we observe a similar effect when increasing the attention length during evaluation.