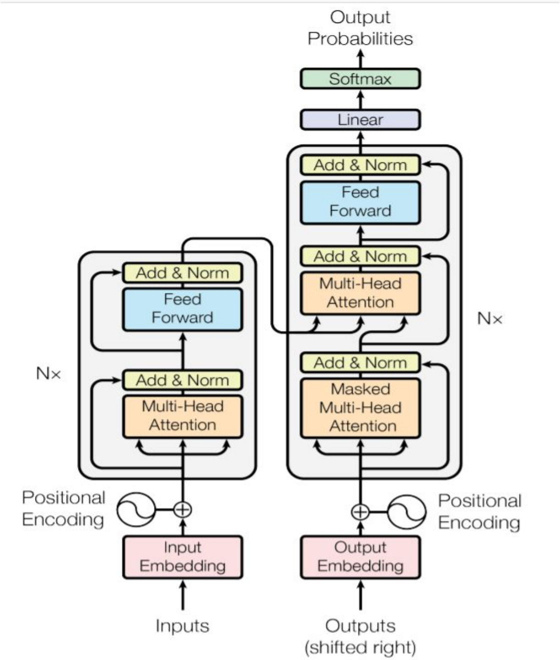
Трансформеры

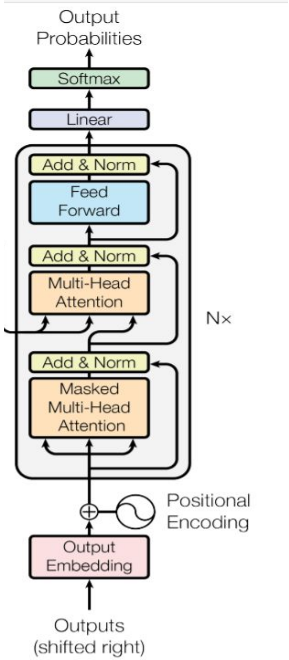
и их улучшения

Акулов Дмитрий
ПМИ 192

Модель Трансформеров

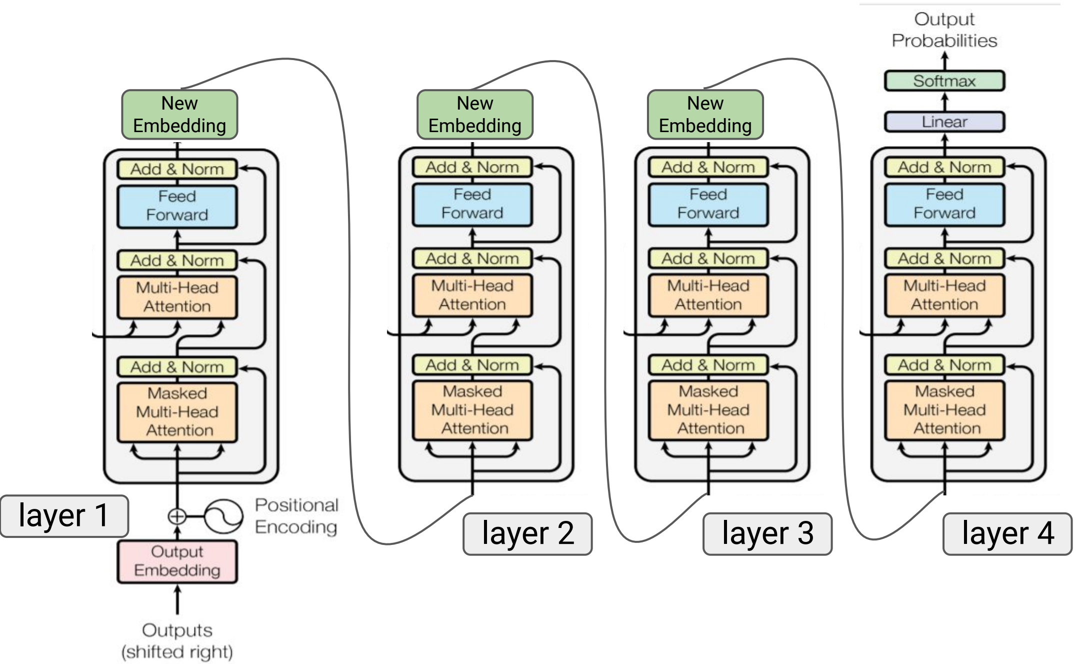


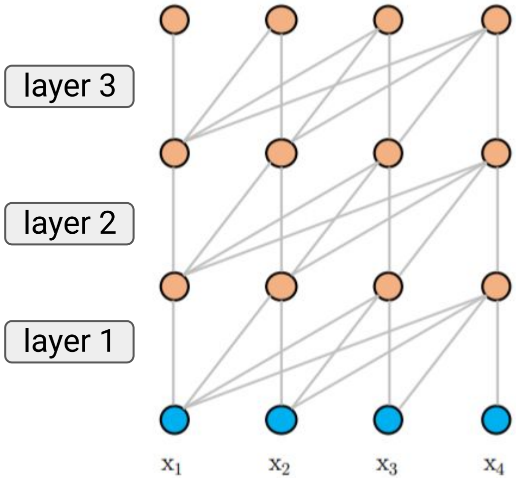
Перевод



Генерация

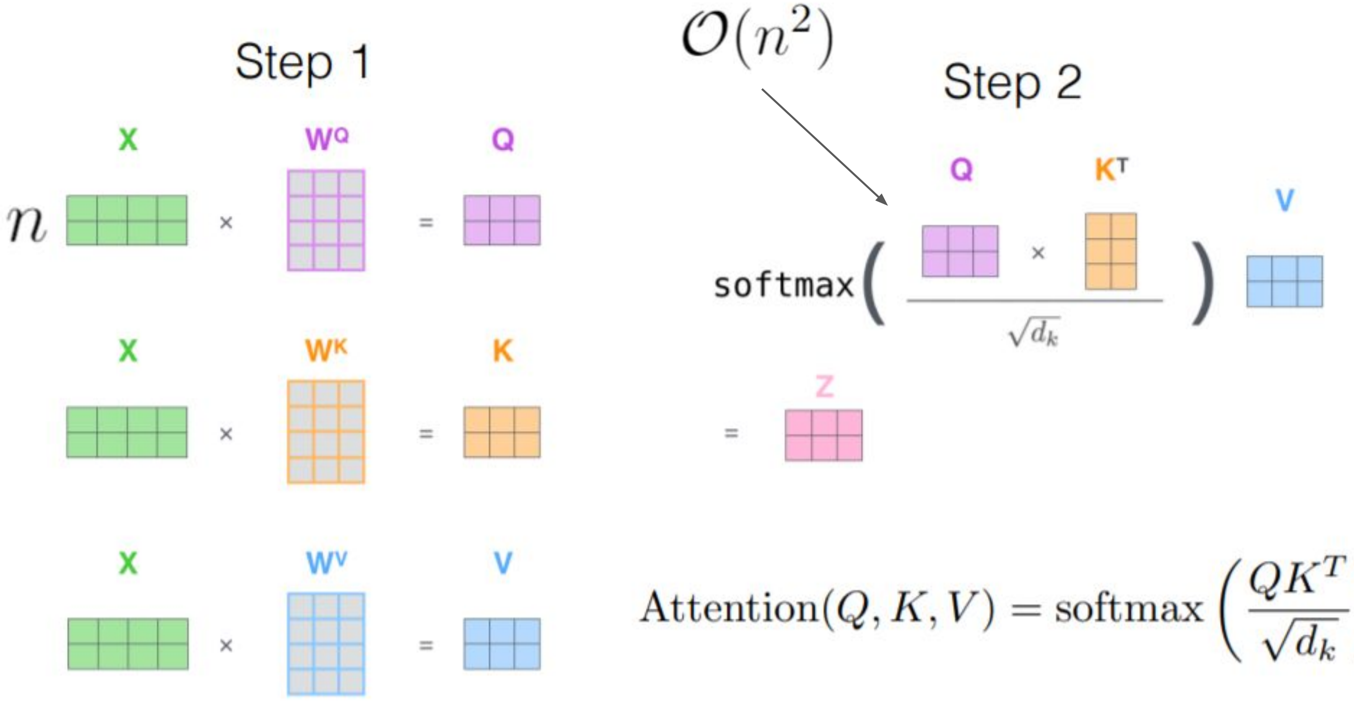
Модель Трансформеров

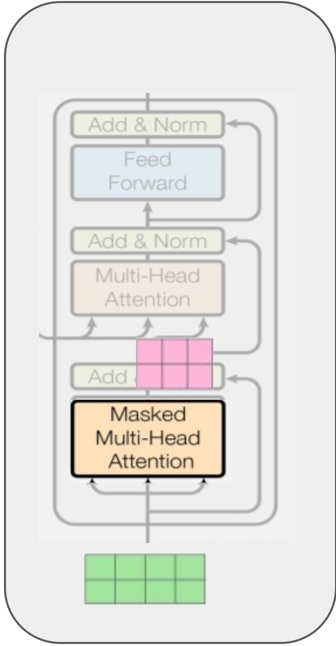




Генерация

Self-Attention

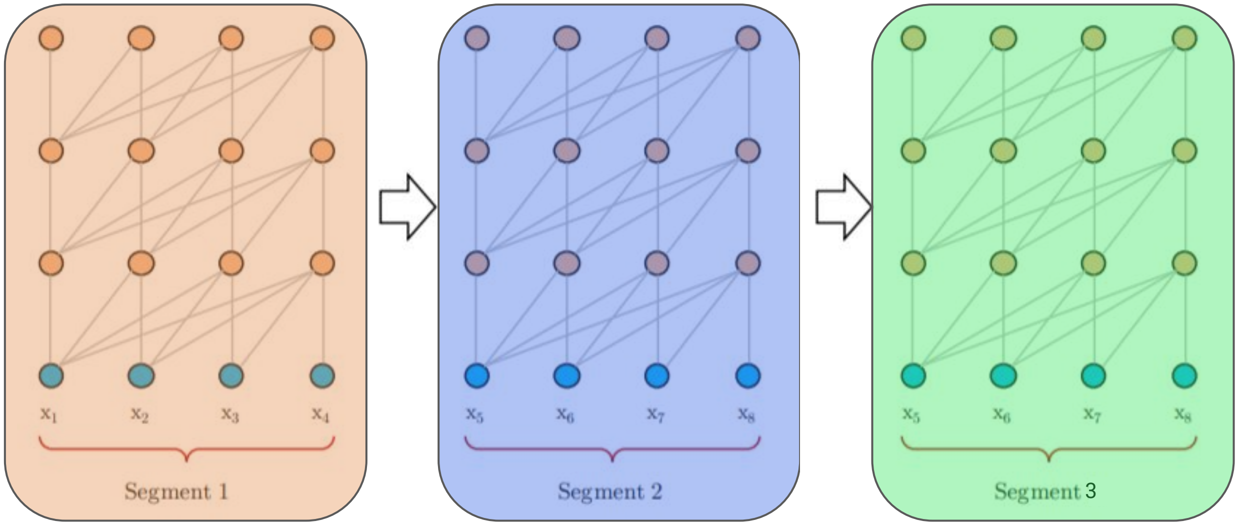




$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Процесс обучения

While we found the idea presented in the previous subsection very appealing

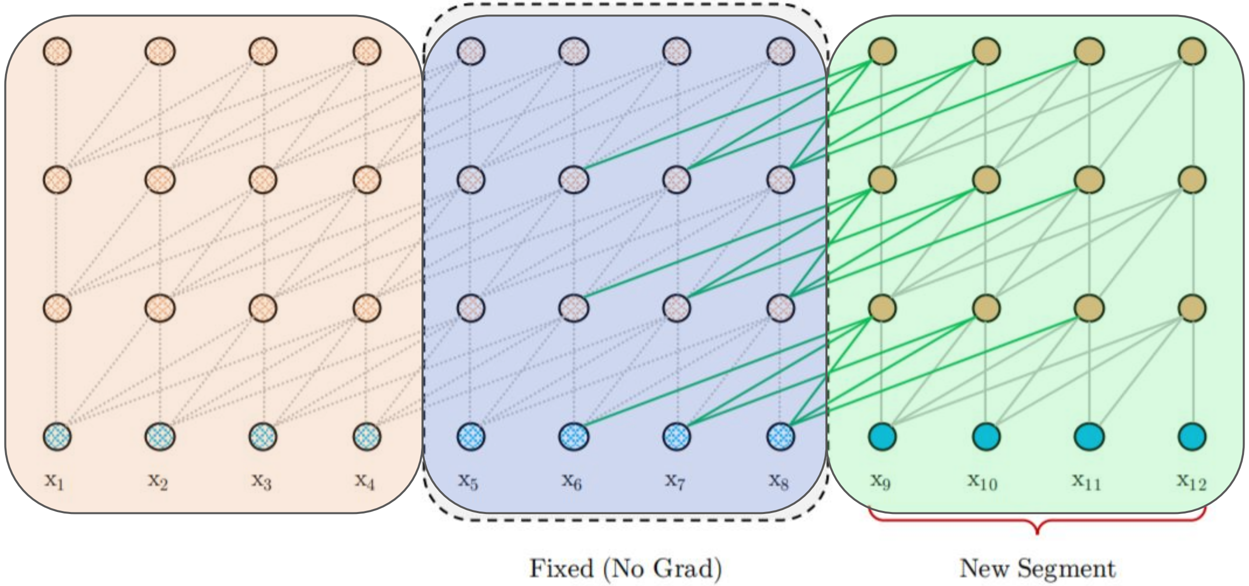


(a) Train phase.

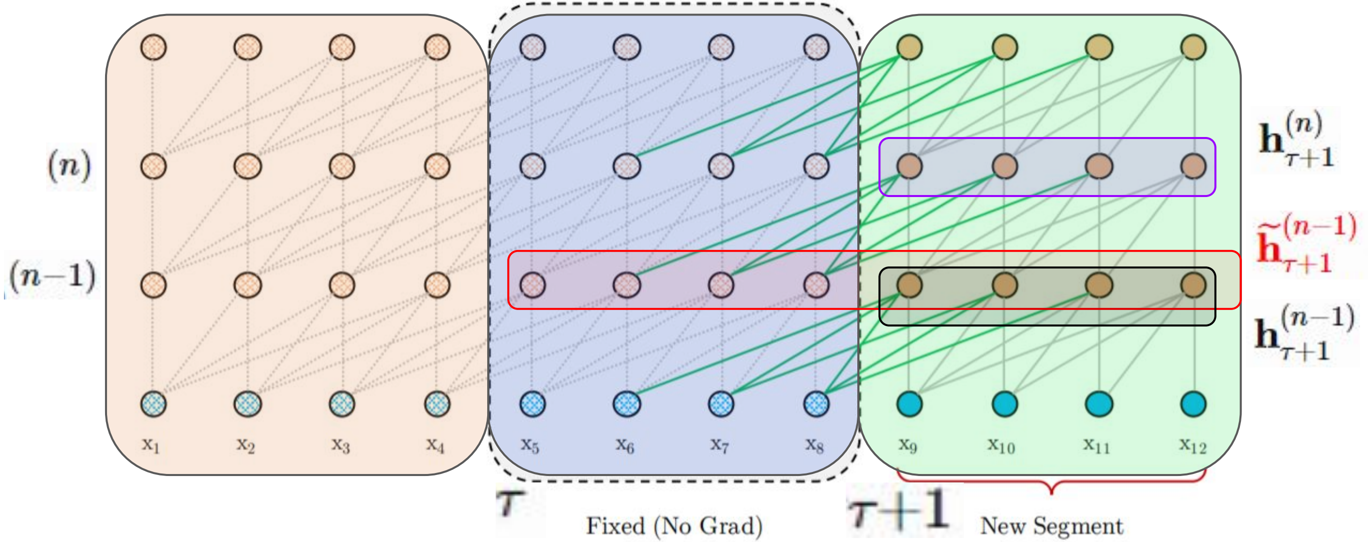
$$n = 12$$

Процесс обучения

While we found the idea presented in the previous subsection very appealing



Transformer-XL



$$\widetilde{\mathbf{h}}_{\tau+1}^{(n-1)} = \left[\text{stop-gradient}(\mathbf{h}_{\tau}^{(n-1)}) \circ \mathbf{h}_{\tau+1}^{(n-1)}\right]$$

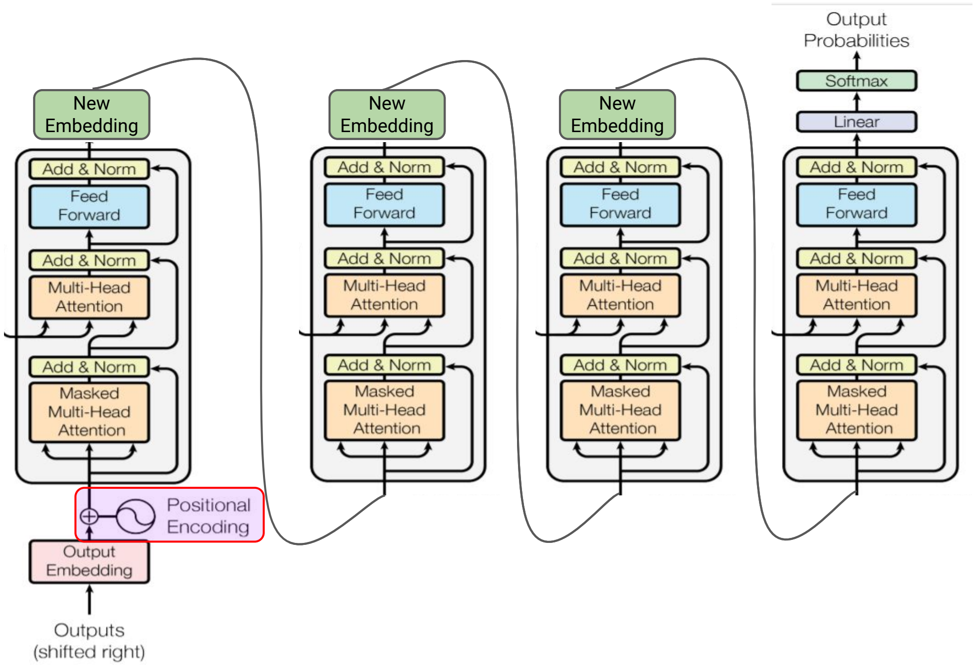
$$\mathbf{Q}_{\tau+1}^{(n)} = \mathbf{h}_{\tau+1}^{(n-1)} \mathbf{W}^{q}$$

$$\mathbf{K}_{\tau+1}^{(n)} = \widetilde{\mathbf{h}}_{\tau+1}^{(n-1)} \mathbf{W}^{k}$$

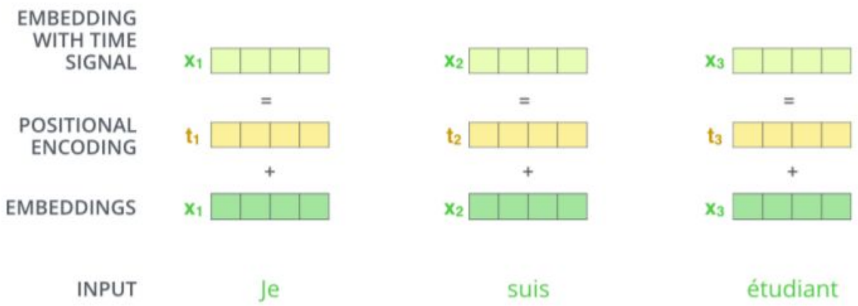
$$\mathbf{V}_{\tau+1}^{(n)} = \widetilde{\mathbf{h}}_{\tau+1}^{(n-1)} \mathbf{W}^{v}$$

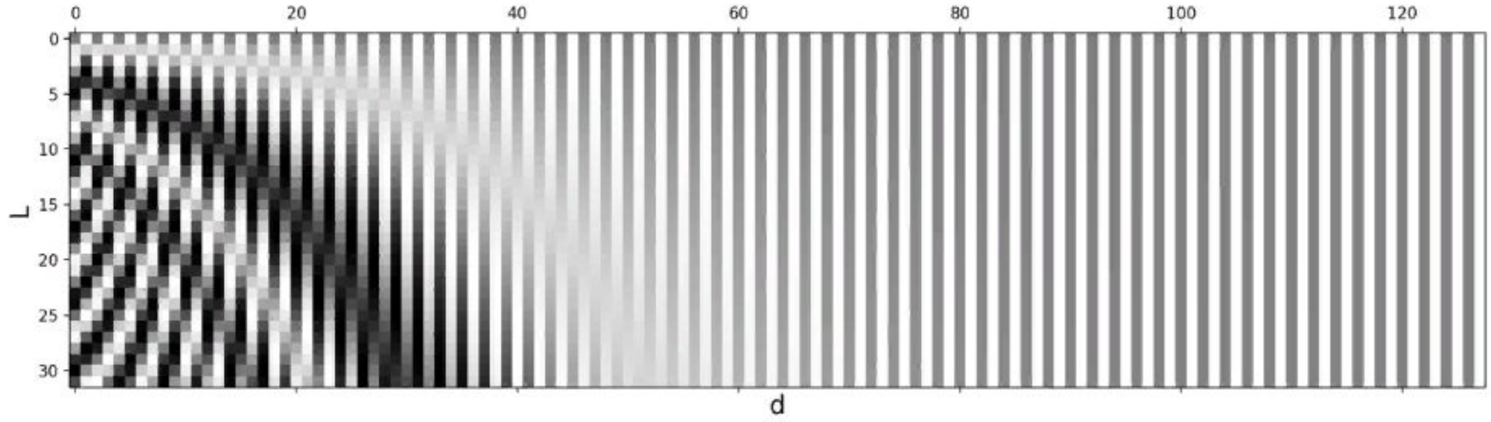
$$\mathbf{h}_{\tau+1}^{(n)} = \text{transformer-layer}(\mathbf{Q}_{\tau+1}^{(n)}, \mathbf{K}_{\tau+1}^{(n)}, \mathbf{V}_{\tau+1}^{(n)})$$

Positional Encoding

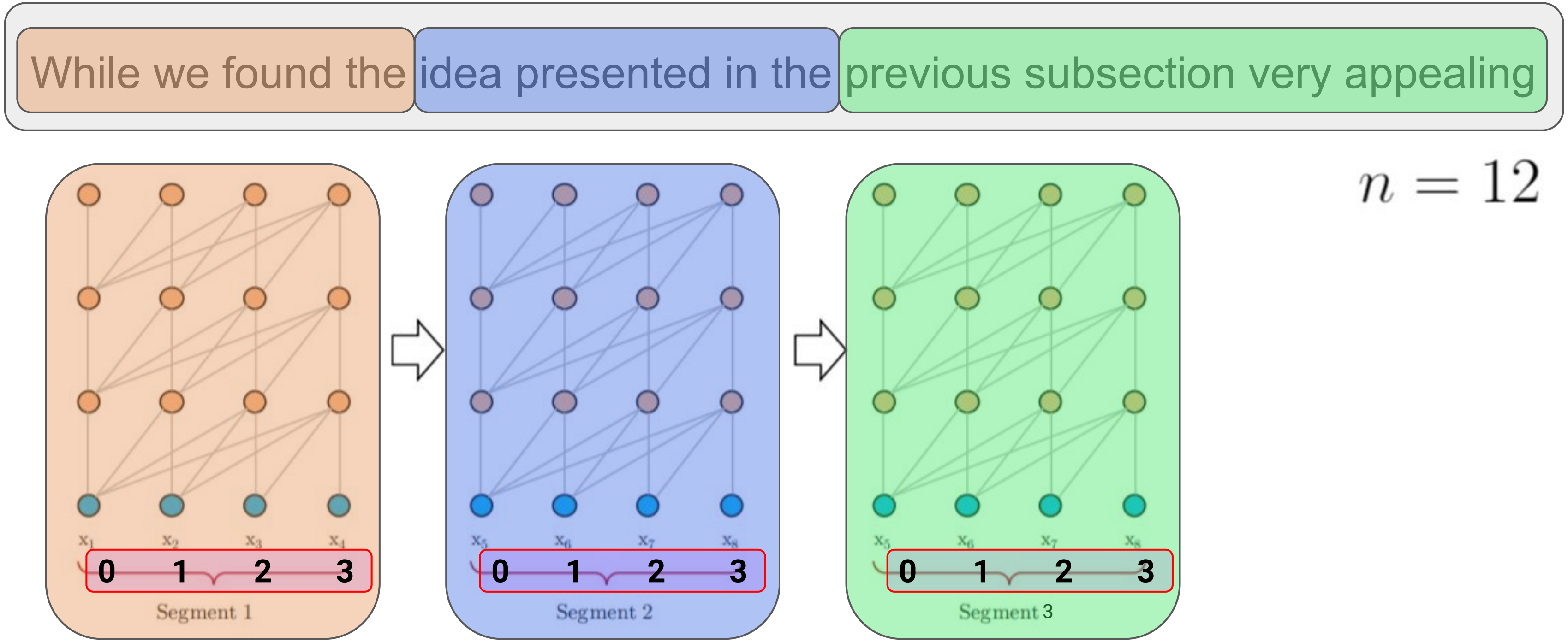


Positional Encoding





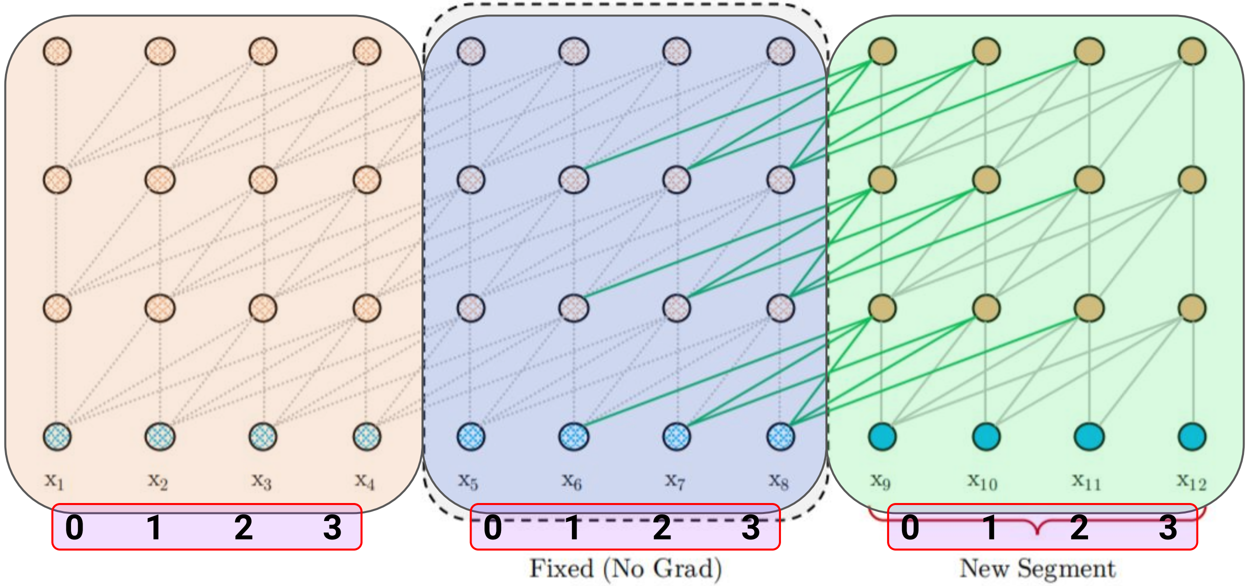
Positional Encoding



(a) Train phase.

Positional Encoding

While we found the | idea presented in the | previous subsection very appealing



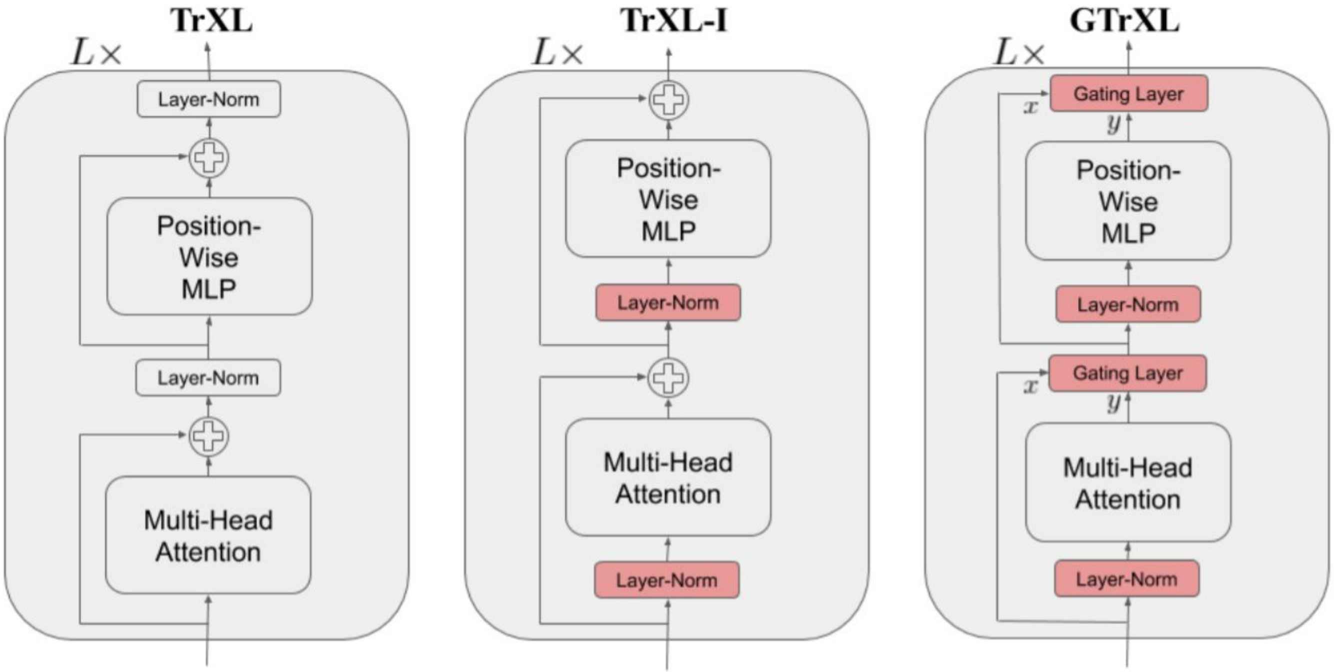
Positional Encoding

$$
\begin{aligned}
a_{ij} = \mathbf{q}_i \mathbf{k}_j^\top &= (\mathbf{x}_i + \mathbf{p}_i)\mathbf{W}^q((\mathbf{x}_j + \mathbf{p}_j)\mathbf{W}^k)^\top \\
&= \mathbf{x}_i \mathbf{W}^q {\mathbf{W}^k}^\top \mathbf{x}_j^\top + \mathbf{x}_i \mathbf{W}^q {\mathbf{W}^k}^\top \mathbf{p}_j^\top + \mathbf{p}_i \mathbf{W}^q {\mathbf{W}^k}^\top \mathbf{x}_j^\top + \mathbf{p}_i \mathbf{W}^q {\mathbf{W}^k}^\top \mathbf{p}_j^\top
\end{aligned}
$$

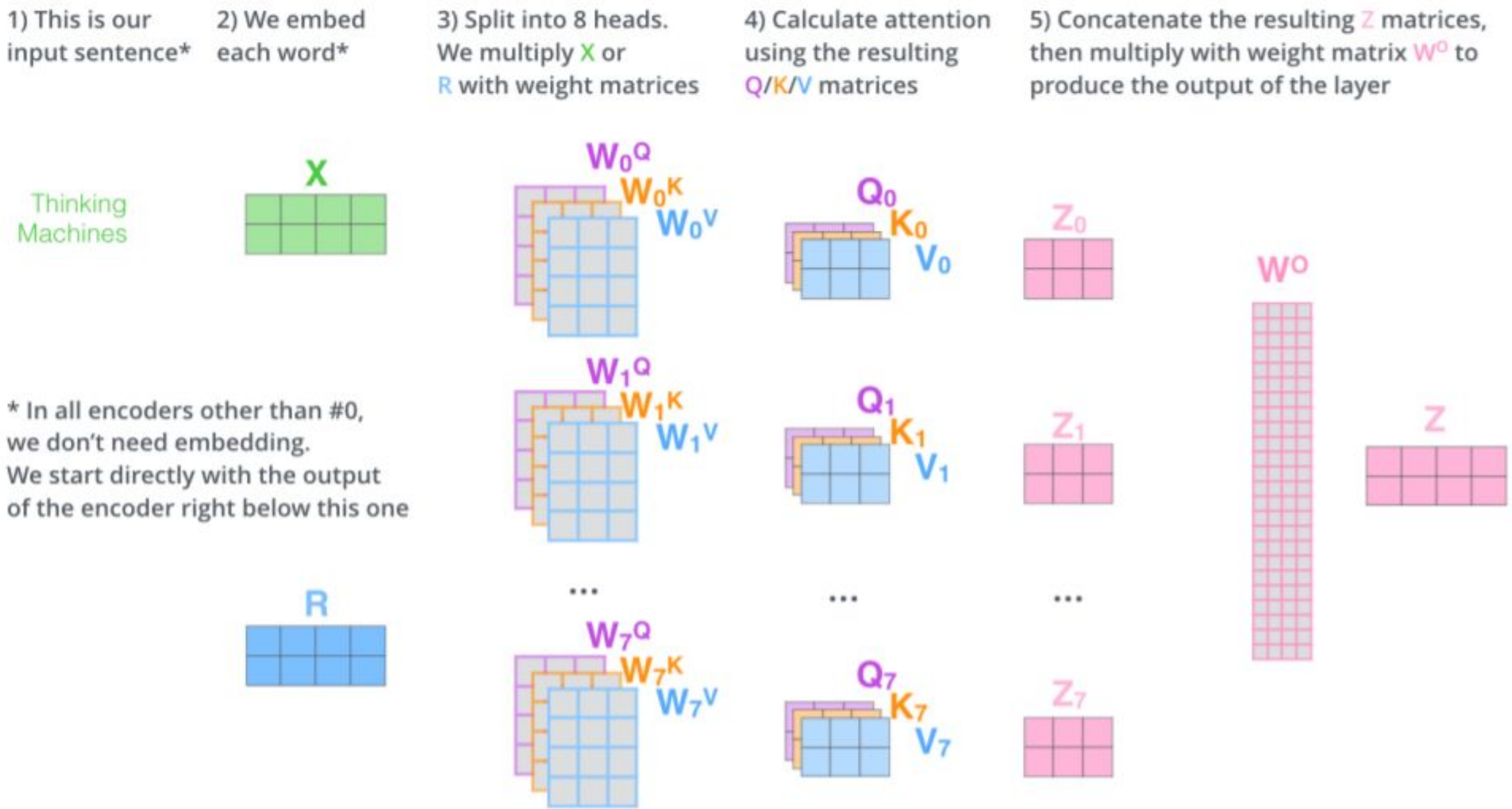
Transformer-XL reparameterizes the above four terms as follows:

$$
a_{ij}^{\text{rel}} = \underbrace{\mathbf{x}_i \mathbf{W}^q {\mathbf{W}_E^k}^\top \mathbf{x}_j^\top}_{\text{content-based addressing}} + \underbrace{\mathbf{x}_i \mathbf{W}^q {\mathbf{W}_R^k}^\top \mathbf{r}_{i-j}^\top}_{\text{content-dependent positional bias}} + \underbrace{\mathbf{u} {\mathbf{W}_E^k}^\top \mathbf{x}_j^\top}_{\text{global content bias}} + \underbrace{\mathbf{v} {\mathbf{W}_R^k}^\top \mathbf{r}_{i-j}^\top}_{\text{global positional bias}}
$$

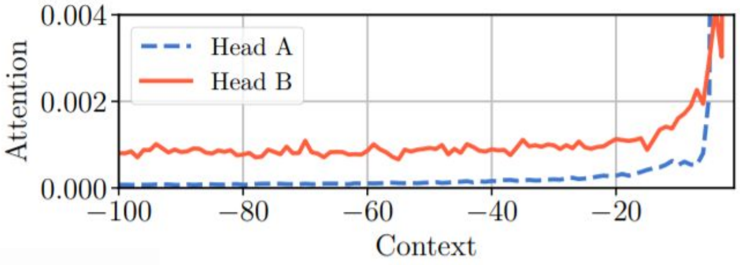
Stabilization for RL (GTrXL)



MultiHead Attention



Adaptive Attention Span



$$m_z(x) = \mathrm{clamp}(\frac{1}{R}(R + z - x), 0, 1)$$

where R is a hyper-parameter which defines the softness of m_z.

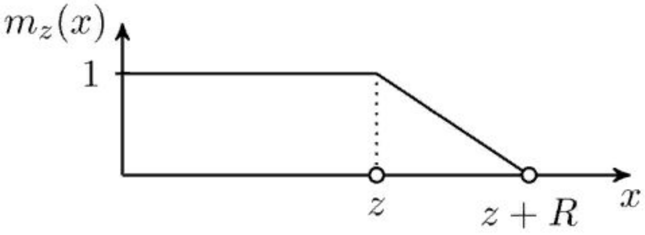
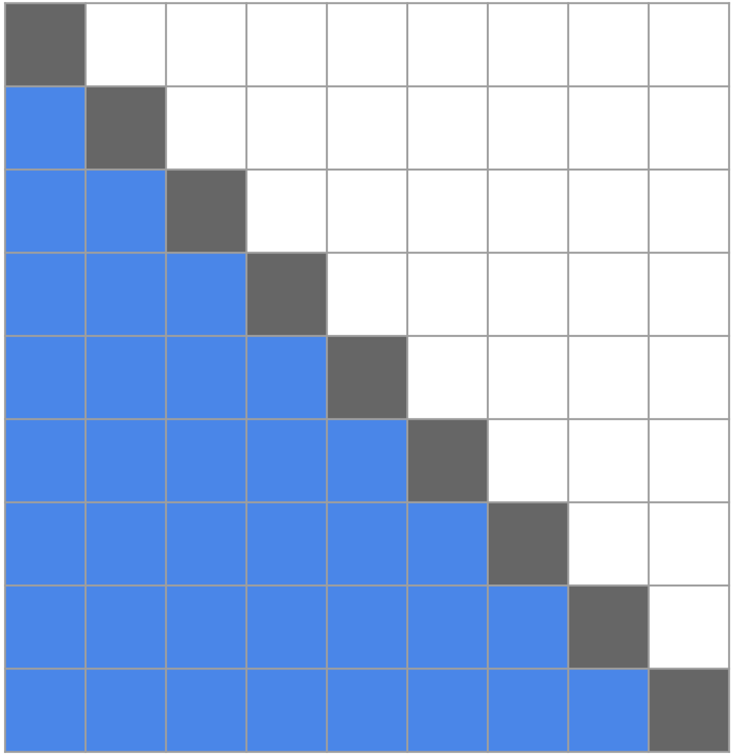


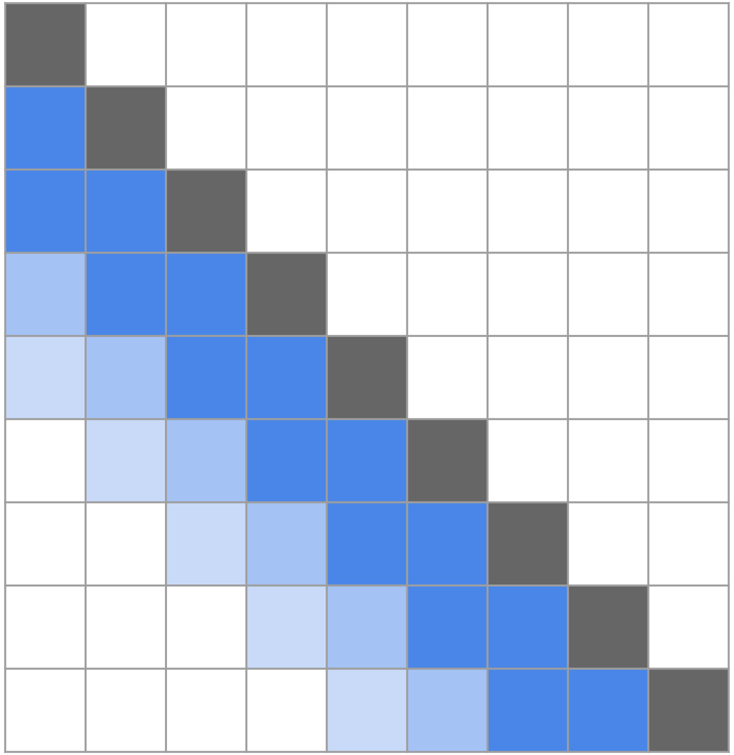
Fig. 8. The soft masking function used in the adaptive attention span. (Image source: Sukhbaatar, et al. 2019.)

The soft mask function is applied to the softmax elements in the attention weights:

$$a_{ij} = \frac{m_z(i-j)\exp(s_{ij})}{\sum_{r=i-s}^{i-1} m_z(i-r)\exp(s_{ir})}$$

Adaptive Attention Span





Результаты

Model	#layers	Avg. span	#Params	#FLOPS	dev	test
Small models						
T12 (Al-Rfou et al., 2019)	12	512	44M	22G	-	1.18
Adaptive-Span ($S = 8192$)	12	314	38M	42M	1.05	**1.11**
Large models						
T64 (Al-Rfou et al., 2019)	64	512	235M	120G	1.06	1.13
T-XL (Dai et al., 2019)	24	3800	277M	438M	-	1.08
Adaptive-Span ($S = 8192$)	24	245	209M	179M	1.01	**1.07**

Литература

- https://lilianweng.github.io/lil-log/2020/04/07/the-transformer-family.html
- https://arxiv.org/abs/1706.03762
- https://arxiv.org/abs/1901.02860
- https://arxiv.org/abs/1910.06764
- https://arxiv.org/abs/1905.07799