DeepNet: Scaling Transformers to 1,000 Layers

Hongyu Wang* **Shuming Ma*** **Li Dong** **Shaohan Huang** **Dongdong Zhang** **Furu Wei†**
Microsoft Research
https://github.com/microsoft/unilm

Background

- The depth of Transformers is limited by its instability.

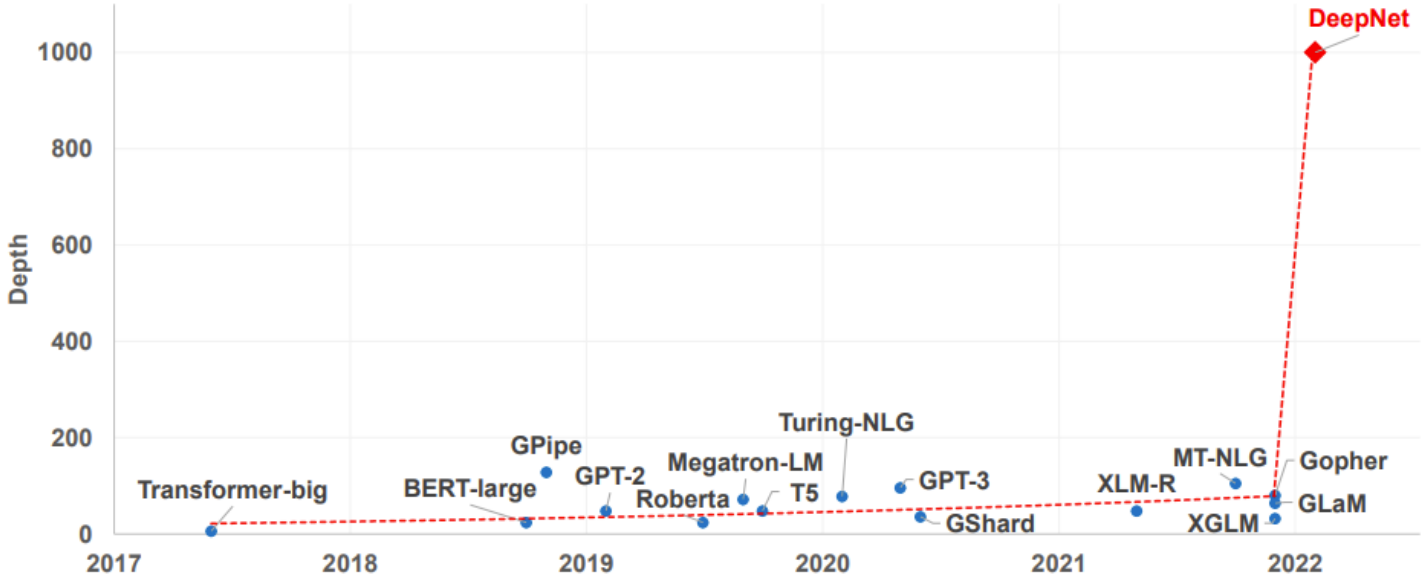
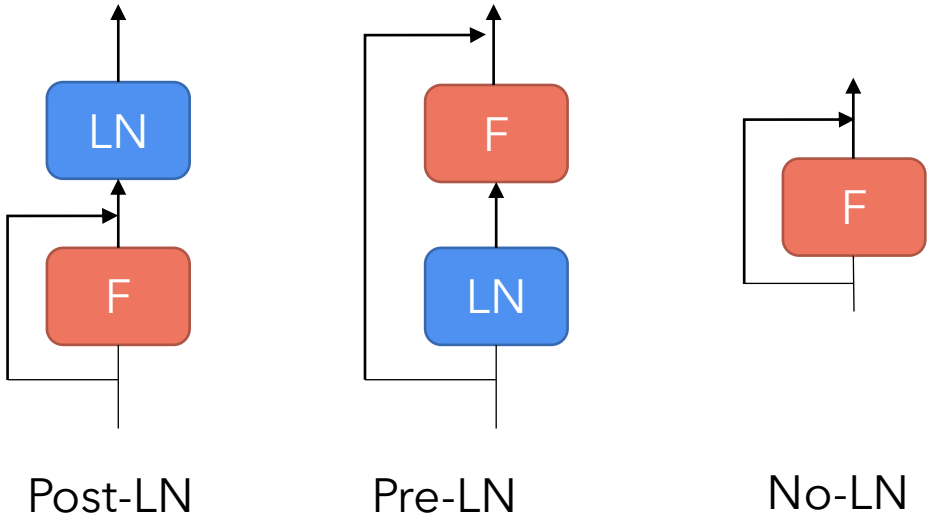


Figure 1: Trend of Transformer depths of state-of-the-art NLP models over time.

Background

- The depth of Transformers is limited by its instability.
- Stable training of Pre-LN & Good performance of Post-LN



Post-LN Pre-LN No-LN

Related Works

- Post-LN
  - Improving Deep Transformer with Depth-Scaled Initialization and Merged Attention (DS-init)
  - Understanding the Difficulty of Training Transformers (Admin)
  - On the Variance of the Adaptive Learning Rate and Beyond (RAdam)
- Pre-LN
  - NormFormer: Improved Transformer Pretraining with Extra Normalization (NormFormer)
  - Transformers without Tears: Improving the Normalization of Self-Attention
  - Learning Deep Transformer Models for Machine Translation (DLCL)
- No-LN
  - Fixup Initialization: Residual Learning Without Normalization (R-Fixup)
  - Improving Transformer Optimization Through Better Initialization (T-Fixup)
  - ReZero is All You Need: Fast Convergence at Large Depth (ReZero)
- Theory:
  - On Layer Normalization in the Transformer Architecture
  - Visualizing the Loss Landscape of Neural Nets

Agenda

- Overview
- Instability of Deep Transformer
- DeepNet
  - Bound of Model Update
  - DeepNorm and its initialization
- Experiments
  - NMT
  - Multilingual NMT
- Future work

Overview

- Model Update $||\Delta F|| = ||F(x, \theta^*) - F(x, \theta)||$. We introduce DeepNorm at residual connections to bound $||\Delta F||$ by a constant.

- DeepNorm significantly improves stability of Post-LN Transformer so that we can easily scale it to >1,000L.

- 200-layer DeepNet with 3.2B param (deep and thin) significantly outperforms the 48-layer M2M-100 with 12B param (shallow and wide) by 5 BLEU on a multilingual benchmark with 7,482 translation directions.

Overview

```
def deepnorm(x):
    return LayerNorm(x * α + f(x))

def deepnorm_init(w):
    if w is ['ffn', 'v_proj', 'out_proj']:
        nn.init.xavier_normal_(w, gain=β)
    elif w is ['q_proj', 'k_proj']:
        nn.init.xavier_normal_(w, gain=1)
```

Architectures	Encoder		Decoder	
	α	β	α	β
Encoder-only (e.g., BERT)	$(2N)^{\frac{1}{4}}$	$(8N)^{-\frac{1}{4}}$	-	-
Decoder-only (e.g., GPT)	-	-	$(2M)^{\frac{1}{4}}$	$(8M)^{-\frac{1}{4}}$
Encoder-decoder (e.g., NMT, T5)	$0.81(N^4M)^{\frac{1}{16}}$	$0.87(N^4M)^{-\frac{1}{16}}$	$(3M)^{\frac{1}{4}}$	$(12M)^{-\frac{1}{4}}$

Figure 2: (a) Pseudocode for DEEPNORM. We take Xavier initialization (Glorot and Bengio, 2010) as an example, and it can be replaced with other standard initialization. Notice that α is a constant. (b) Parameters of DEEPNORM for different architectures (N-layer encoder, M-layer decoder).

Overview

- Model update of Post-LN grows problematically fast as depth increases. DeepNorm bounds model update independent of depth.

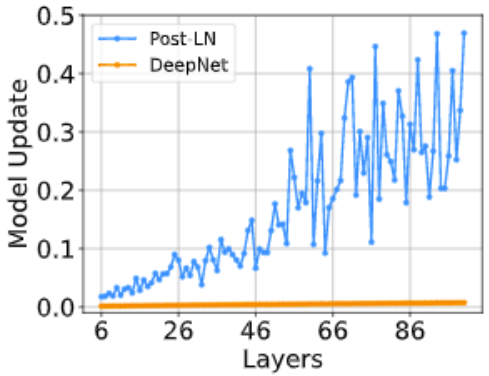


Figure 5: Model updates of vanilla Post-LN and DEEPNET at the early stage of training. The visualization is conducted on 64-128-2 tiny Transformers with depth varying from 6L-6L to 100L-100L. It shows that DEEPNET has much smaller and more stable updates than Post-LN.

Instability of Deep Transformer

- Gradient exploding ?
  - Post-LN: $W_{ij} \sim N\left(0, \frac{2}{d_{in}+d_{out}}\right)$
  - Post-LN-init: $W_{ij} \sim N\left(0, \frac{2}{k_l^2(d_{in}+d_{out})}\right)$, $k_l = N - l + 1$

Instability of Deep Transformer

- Gradient exploding ?

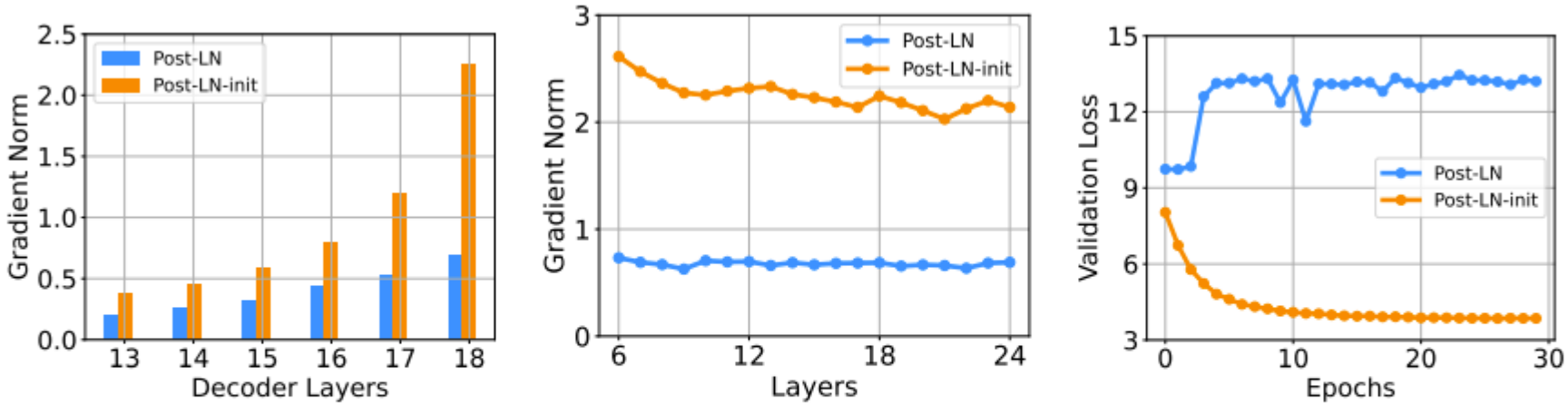


Figure 3: (a) Gradient norm in the top layers of 18L-18L models. (b) Gradient norm in the last layer of the models with depths varying from 6L-6L to 24L-24L. (c) Validation loss curves of 18L-18L models.

Instability of Deep Transformer

- Why Post-LN-init is successfully trained ?

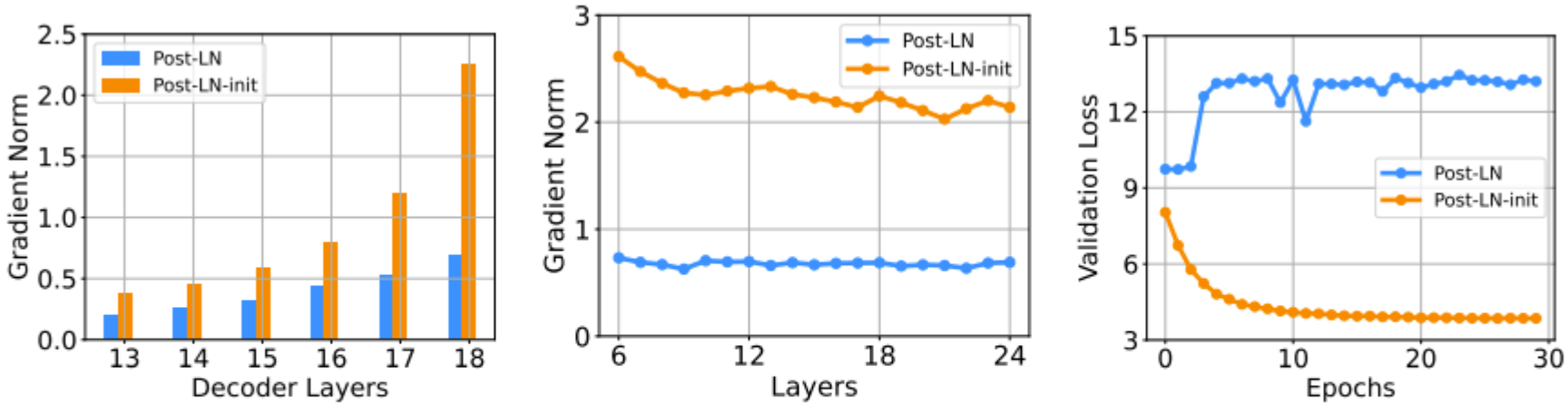
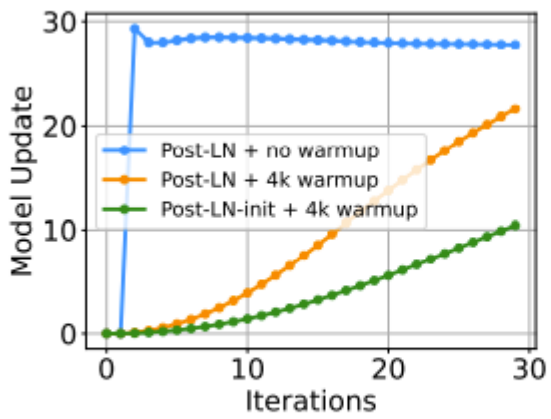


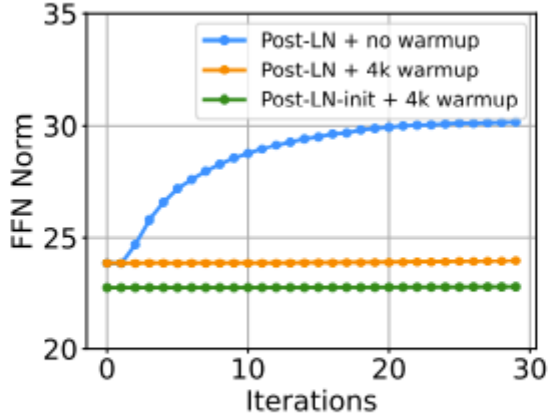
Figure 3: (a) Gradient norm in the top layers of 18L-18L models. (b) Gradient norm in the last layer of the models with depths varying from 6L-6L to 24L-24L. (c) Validation loss curves of 18L-18L models.

Instability of Deep Transformer

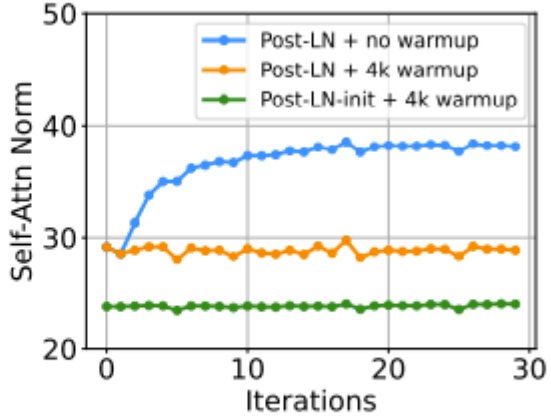
- Exploding Model Update -> Large input to LN



(a) Accumulated model update



(b) Input from FFN to LN

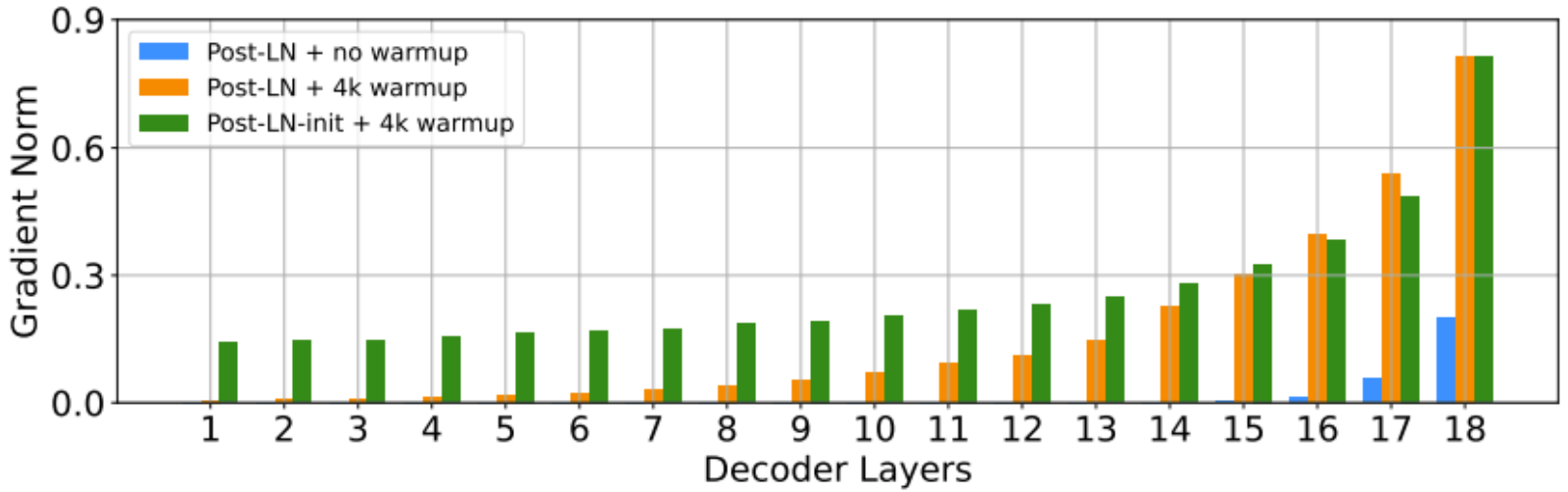


(c) Input from attention to LN

Instability of Deep Transformer

- Large input to LN -> Gradient vanishing

$$||\frac{\partial LN(x)}{\partial x}|| = \mathcal{O}(\frac{\sqrt{d}}{||x||}).$$



(d) Gradient norm in all decoder layers

DeepNet

- Query & Key do not change bound of attention's norm

Lemma 4.1. *Given* $\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, ...\mathbf{x_n})^T \in \mathbf{R}^{n \times d}$, *where* $Var[\mathbf{x_i}] = 1$, $Mean[\mathbf{x_i}] = 0$ *and* $q_i \in \mathbf{R}$ *for all* $i \in [1, n]$, *it satisfies that*

$$softmax(q_1, q_2, ..., q_n)\mathbf{X} \overset{\Theta}{=} \mathbf{x_i},$$

where $\overset{\Theta}{=}$ *stands for equal bound of magnitude.*[3]

$$Attn(Q, K, V) \overset{\Theta}{=} vwV \qquad FFN(X) \overset{\Theta}{=} vwX$$

DeepNet

- Assumption:

1. Hidden dimension d equals to 1.
2. $Var[x + G_l(x)] \overset{\Theta}{=} Var[x] + Var[G_l(x)]$
3. All relevant weights v, w are positive with magnitude less than 1 and α, β for DEEPNORM are positive with magnitude greater than 1.

- Self-Attn & FFN

$$x_{l+1} = f_l(x_l, \theta_l) = \frac{\alpha x_l + G_l(x_l)}{\sqrt{Var[\alpha x_l + G_l(x_l)]}} \overset{\Theta}{=} \frac{\alpha + v_l w_l}{\sqrt{\alpha^2 + v_l^2 w_l^2}} x_l$$

DeepNet

$$||\Delta F|| = ||F(x, \theta^*) - F(x, \theta)|| = ||x^*_{2N+1} - x_{2N+1}|| = ||f(x^*_{2N}, \theta^*_{2N}) - f(x_{2N}, \theta_{2N})|| \quad (6)$$

Using Taylor expansion for Equation (6), we get:

$$\begin{aligned}
||\Delta F|| &= ||x^*_{2N+1} - x_{2N+1}|| \quad (7)\\
&\approx ||\frac{\partial f}{\partial x}(x_{2N}, \theta_{2N})(x^*_{2N} - x_{2N}) + \frac{\partial f}{\partial \theta}(x_{2N}, \theta_{2N})(\theta^*_{2N} - \theta_{2N})^T||\\
&\leq ||\frac{\partial f}{\partial x}(x_{2N}, \theta_{2N})|| \cdot ||x^*_{2N} - x_{2N}|| + ||\frac{\partial f}{\partial \theta}(x_{2N}, \theta_{2N})|| \cdot ||\theta^*_{2N} - \theta_{2N}||\\
&= \frac{\alpha + v_{2N}w_{2N}}{\sqrt{\alpha^2 + v^2_{2N}w^2_{2N}}}||x^*_{2N} - x_{2N}|| + \frac{\alpha(\alpha - v_{2N}w_{2N})}{(\alpha^2 + v^2_{2N}w^2_{2N})^{\frac{3}{2}}}\sqrt{v^2_{2N} + w^2_{2N}}||\theta^*_{2N} - \theta_{2N}||\\
&\approx ||x^*_{2N} - x_{2N}|| + \frac{\sqrt{v^2_{2N} + w^2_{2N}}}{\alpha}||\theta^*_{2N} - \theta_{2N}|| \quad (8)
\end{aligned}$$

DeepNet

- Expected bound of Only-Encoder & Only-Decoder

Theorem 4.2. *Given an N-layer* DEEPNET $F(x, \theta)$ *($\theta = \{\theta_1, \theta_2, ..., \theta_{2N}\}$), where θ_{2l-1} and θ_{2l} denote the parameters of self-attention and FFN in l-th layer, and each sub-layer is normalized with* DEEPNORM*: $x_{l+1} = LN(\alpha x_l + G_l(x_l, \theta_l))$, $||\Delta F||$ satisfies:*

$$||\Delta F|| \leq \sum_{i=1}^{2N} \frac{\sqrt{v_i^2 + w_i^2}}{\alpha} ||\theta_i^* - \theta_i||$$

DeepNet

- Cross-Attn Update:

$$
\begin{aligned}
||y^*_{l+1} - y_{l+1}|| &= ||f^*_{dl}(y^*_l, x^*_{2N+1}, \theta^*_{dl}) - f_{dl}(y_l, x_{2N+1}, \theta_{dl})|| \\
&\approx \frac{\alpha_d}{\sqrt{\alpha_d^2 + v_{dl}^2 w_{dl}^2}} ||y^*_l - y_l|| + \frac{v_{dl} w_{dl}}{\sqrt{\alpha_d^2 + v_{dl}^2 w_{dl}^2}} ||x^*_{2N+1} - x_{2N+1}|| \\
&\quad + \frac{\alpha_d(\alpha_d - v_{dl} w_{dl})}{(\alpha_d^2 + v_{dl}^2 w_{dl}^2)^{\frac{3}{2}}} \sqrt{v_{dl}^2 + w_{dl}^2} ||\theta^*_{dl} - \theta_{dl}|| \\
&\leq ||y^*_l - y_l|| + \frac{v_{dl} w_{dl}}{\alpha_d} ||x^*_{2N+1} - x_{2N+1}|| + \frac{\sqrt{v_{dl}^2 + w_{dl}^2}}{\alpha_d} ||\theta^*_{dl} - \theta_{dl}|| \qquad (12)
\end{aligned}
$$

- Encoder Update is propagated to decoder through Cross-Attn

DeepNet

- Expected bound of Encoder-Decoder

Theorem 4.3. *Given an encoder-decoder* DEEPNET $F_{ed}(x, y, \theta_e, \theta_d)$ *with N encoder layers and M decoder layers, where each encoder sub-layer is normalized as* $x_{l+1} = LN(\alpha_e x_l + G_{el}(x_l, \theta_{el}))$, *and the decoder sub-layer is normalized as* $x_{l+1} = LN(\alpha_d x_l + G_{dl}(x_l, \theta_{dl}))$, $||\Delta F_{ed}||$ *satisfies:*

$$||\Delta F_{ed}|| \leq \sum_{j=1}^{M} \frac{v_{d,3j-1} w_{d,3j-1}}{\alpha_d} \sum_{i=1}^{2N} \frac{\sqrt{v_{ei}^2 + w_{ei}^2}}{\alpha_e} ||\theta_{ei}^* - \theta_{ei}|| + \sum_{j=1}^{3M} \frac{\sqrt{v_{dj}^2 + w_{dj}^2}}{\alpha_d} ||\theta_{dj}^* - \theta_{dj}|| \quad (1)$$

DeepNet

- Our goal for model update:

GOAL: $F_{ed}(x, y, \theta_e, \theta_d)$ is updated by $\Theta(\eta)$ per SGD step after initialization as $\eta \to 0$. That is $||\Delta F_{ed}|| = \Theta(\eta)$ where $\Delta F_{ed} \triangleq F_{ed}(x, y, \theta_e - \eta\frac{\partial \mathcal{L}}{\partial \theta_e}, \theta_d - \eta\frac{\partial \mathcal{L}}{\partial \theta_d}) - F_{ed}(x, y, \theta_e, \theta_d)$.

DeepNet

Encoder-decoder architecture

1. Apply standard initialization (e.g., Xavier initialization) for each encoder and decoder layer.
2. For encoder layers, scale the weights of feed-forward networks as well as the value projection and the output projection of attention layers by $0.87(N^4M)^{-\frac{1}{16}}$, and set the weight of residual connections as $0.81(N^4M)^{\frac{1}{16}}$.
3. For decoder layers, scale the weights of feed-forward networks as well as the value projection and the output projection of attention layers by $(12M)^{-\frac{1}{4}}$, and set the weight of residual connections as $(3M)^{\frac{1}{4}}$.

Encoder-only (or decoder-only) architecture

1. Apply standard initialization (e.g., Xavier initialization) for each layer.
2. For each layer, scale the weights of feed-forward networks as well as the value projection and the output projection of attention layers by $(8N)^{-\frac{1}{4}}$ (or $(8M)^{-\frac{1}{4}}$), and set the weight of residual connections as $(2N)^{\frac{1}{4}}$ (or $(2M)^{\frac{1}{4}}$).

DeepNet

- Model update of DeepNet is nearly constant

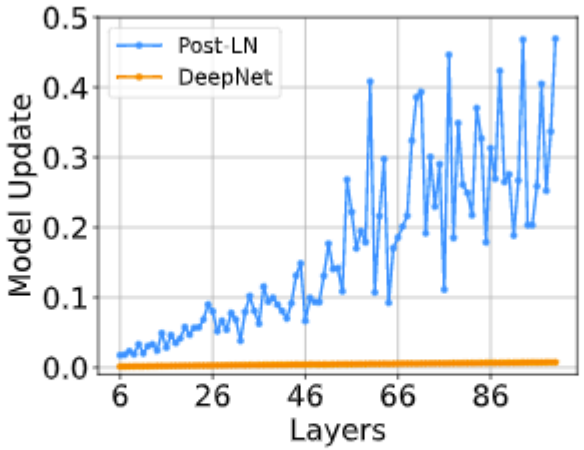


Figure 5: Model updates of vanilla Post-LN and DEEPNET at the early stage of training. The visualization is conducted on 64-128-2 tiny Transformers with depth varying from 6L-6L to 100L-100L. It shows that DEEPNET has much smaller and more stable updates than Post-LN.

Experiments

- Performance
  Deep > Post > Pre > No
- Stability
  Deep ≈ Pre > No > Post

Models	LN	6L-6L	18L-18L	50L-50L	100L-100L
Vanilla Post-LN (Vaswani et al., 2017)	Post	**28.1**		diverged	
DS-Init (Zhang et al., 2019a)	Post	27.9		diverged	
Admin (Liu et al., 2020)	Post	27.9	**28.8**	diverged	
ReZero (Bachlechner et al., 2020)	No	26.9		diverged	
R-Fixup (Zhang et al., 2019b)	No	27.5	28.4	27.7	diverged
T-Fixup (Huang et al., 2020)	No	27.5	28.4	27.9	diverged
Vanilla Pre-LN (Vaswani et al., 2017)	Pre	27.0	28.1	28.0	27.4
DLCL (Wang et al., 2019)	Pre	27.4	28.2	diverged	27.5
NormFormer (Shleifer et al., 2021)	Pre	27.0	28.3	27.8	diverged
DEEPNET (ours)	Deep	27.8	**28.8**	**29.0**	**28.9**

Table 1: BLEU scores on the WMT-17 En-De test set for different models with varying depth. AL-BL refers to A-layer encoder and B-layer decoder.

Experiments

- Compared with Post/No/Pre-LN
  - ReZero is very unstable with Fp16, large lr and droupout

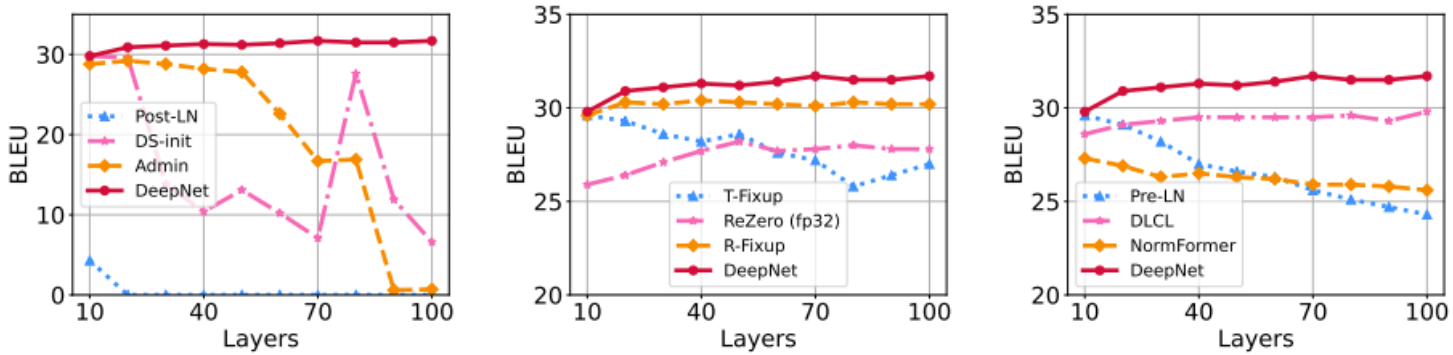


Figure 6: BLEU scores on the IWSLT-14 De-En test set for different deep models with varing depth from 10L-10L to 100L-100L.

Experiments

- Benefit from larger settings, faster & better

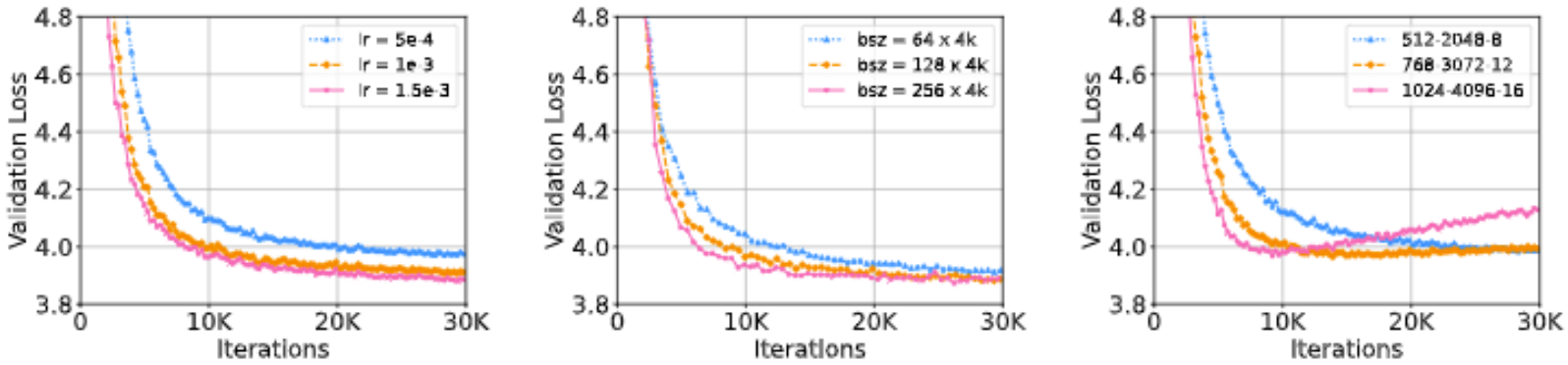


Figure 7: WMT-17 En-De validation loss curves for 18L-18L DEEPNET with varing learning rate, batch size and hidden dimension.

Experiments

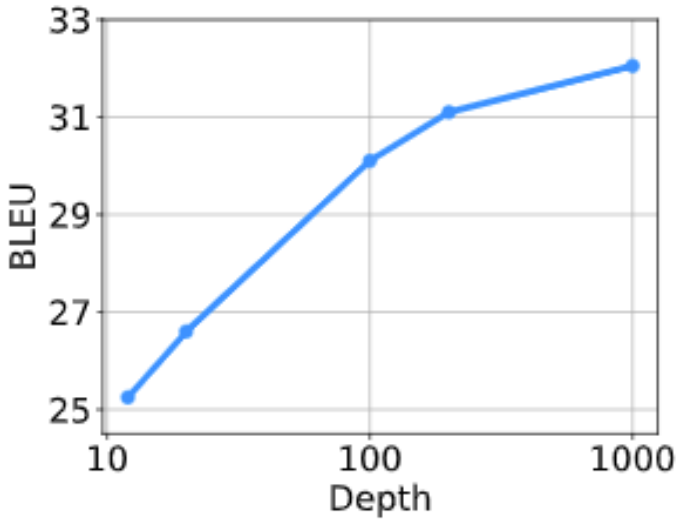
- Scaling to 1,000 Layers !

Models	# Layers	# Params	X→En	En→X	Avg
Baseline (Zhang et al., 2020)	12	133M	27.5	21.4	24.5
	24	173M	29.5	22.9	26.2
	48	254M	31.4	24.0	27.7
DEEPNET (ours)	200	863M	33.2	29.0	31.1
	1000	3.8B	**33.9**	**30.2**	**32.1**

Table 2: Average BLEU for DEEPNET and the baseline on the OPUS-100 test sets.

Experiments

- DeepNet with {12, 20, 100, 200, 1000} layers on OPUS-100
- Scaling law: $L(d) = A\log(d) + B, d$ is model depth



Experiments

- Compared with SOTA model M2M-100
  - Training Data: CCMatrix, CCAligned, OPUS, Tatoeba
  - 5 BLEU improvement on Flores (7,482 directions)

Models	# Layers	# Params	WMT	OPUS	TED	Flores
M2M-100 (Fan et al., 2021)	48	12B	31.9	18.4	18.7	13.6
DEEPNET (ours)	200	3.2B	**33.9**	**23.0**	**20.1**	**18.6**

Table 3: BLEU scores for DEEPNET and M2M-100 on various evaluation sets.

Thanks