Accelerated training through iterative gradient propagation along the residual path

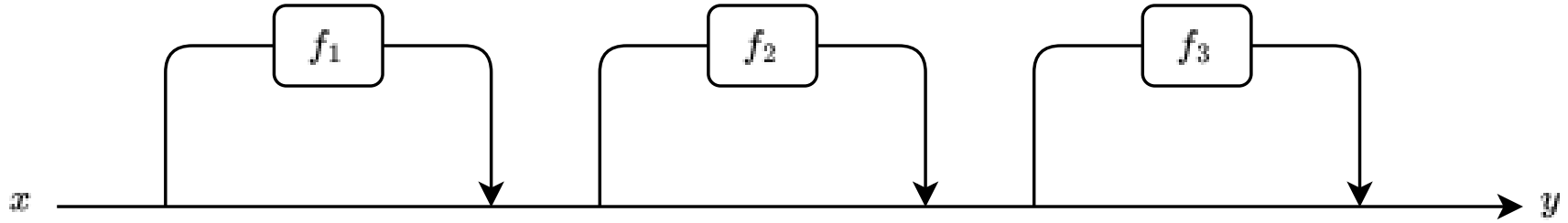
Erwan Fagnou[1], Paul Caillon[1], Blaise Delattre[1,2], Alexandre Allauzen[1,3]

[1] Miles Team, LAMSADE, Université Paris Dauphine-PSL, Paris, France
[2] Foxstream, Vaulx-en-Velin, France
[3] ESPCI PSL, Paris, France

Residual models

- Simplest form: identity residual connection

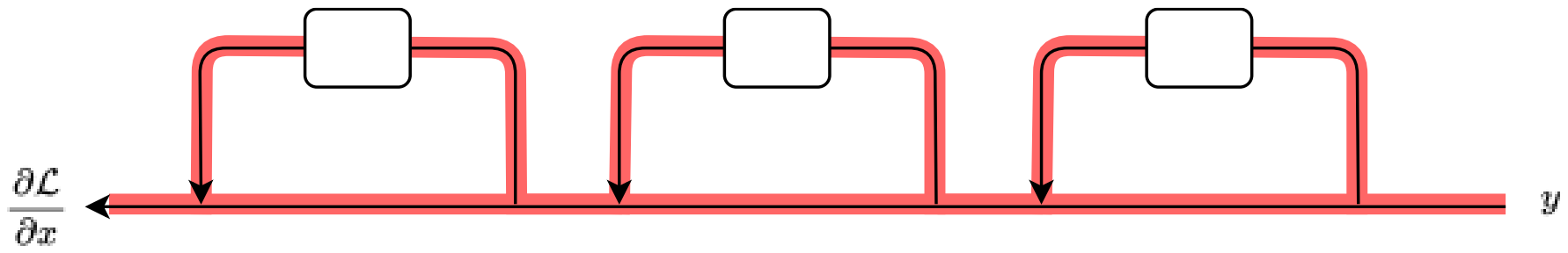
$$h_i = h_{i-1} + f_i(h_{i-1})$$
$$= x + \sum_{j=1}^{i} f_j(h_{j-1})$$



Residual models

- Backpropagation:

$$\frac{\partial \mathcal{L}}{\partial h_i} = \frac{\partial \mathcal{L}}{\partial h_{i+1}}\left(I + J_{j+1}\right)$$
$$= \frac{\partial \mathcal{L}}{\partial h_L}\left(I + J_L\right) \ldots \left(I + J_{i+1}\right)$$



Gradient decomposition

$$\frac{\partial \mathcal{L}}{\partial h_i} = \frac{\partial \mathcal{L}}{\partial h_L}(I + J_L) \dots (I + J_{i+1})$$

Gradient decomposition

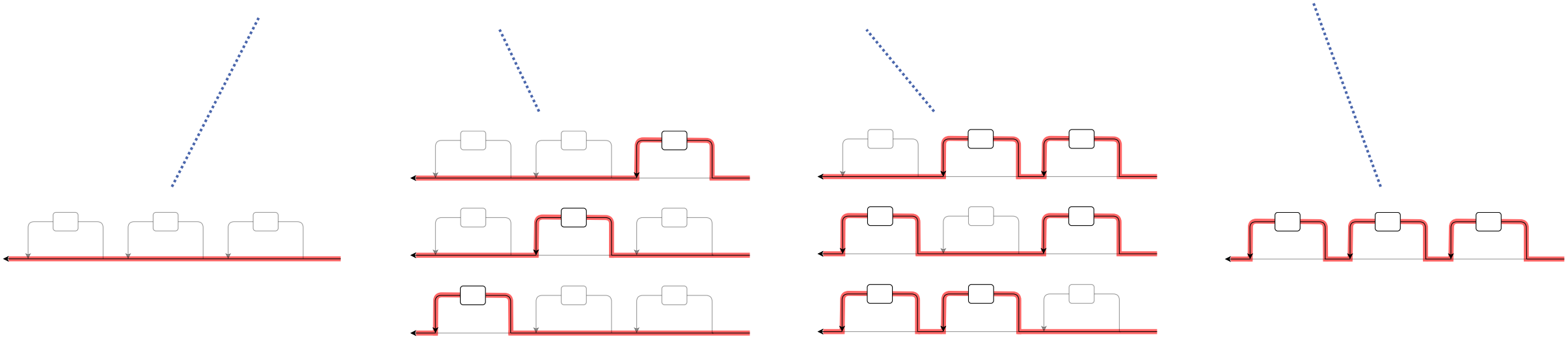
$$\frac{\partial \mathcal{L}}{\partial h_i} = \frac{\partial \mathcal{L}}{\partial h_L}(I + J_L)\dots(I + J_{i+1})$$

$$= \frac{\partial \mathcal{L}}{\partial h_L}\left[I + \sum_{j=i+1}^{L} J_j + \sum_{j=i+1}^{L}\sum_{k=j+1}^{L} J_j J_k + \dots + J_L J_{L-1} \cdots J_{i+1} \right]$$

Gradient decomposition

$$\frac{\partial \mathcal{L}}{\partial h_i} = \frac{\partial \mathcal{L}}{\partial h_L}(I + J_L)\dots(I + J_{i+1})$$

$$= \frac{\partial \mathcal{L}}{\partial h_L}\left[I + \sum_{j=i+1}^{L} J_j + \sum_{j=i+1}^{L}\sum_{k=j+1}^{L} J_k J_j + \dots + J_L J_{L-1} \cdots J_{i+1} \right]$$



Gradient decomposition

True gradient:

$$\frac{\partial \mathcal{L}}{\partial h_i} = \sum_{n=0}^{L} "gradients\ going\ through\ n\ blocks"$$

Approximation with partial sum:

$$w_i^k = \sum_{n=0}^{\boldsymbol{k}} "gradients\ going\ through\ n\ blocks"$$

Gradient decomposition

True gradient:

$$\frac{\partial \mathcal{L}}{\partial h_i} = \sum_{n=0}^{L} "gradients\ going\ through\ n\ blocks"$$

Approximation with partial sum:

$$w_i^k = \sum_{n=0}^{\boldsymbol{k}} "gradients\ going\ through\ n\ blocks"$$

→ Why would this work?

- Gradient vanishing reduces the gradients going through many blocks

Highway backpropagation

Iterative computation:

$$\begin{cases} \mathrm{w}_\mathrm{i}^0 = \sum_{j=i}^{L} \frac{\partial \mathcal{L}_j}{\partial h_j} \\ \mathrm{w}_\mathrm{i}^{\mathrm{k}+1} = w_i^0 + \sum_{j=i+1}^{L} w_j^k J_j \end{cases}$$

Highway backpropagation

Iterative computation:

$$\begin{cases} \mathrm{w}_\mathrm{i}^0 = \sum_{j=i}^{L} \frac{\partial \mathcal{L}_j}{\partial h_j} \\ \mathrm{w}_\mathrm{i}^{k+1} = w_i^0 + \sum_{j=i+1}^{L} w_j^k J_j \end{cases}$$

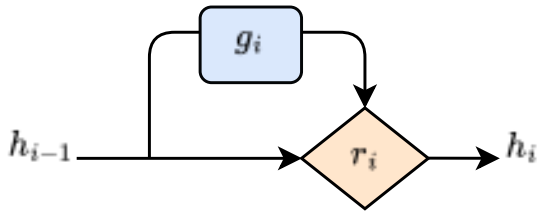
1) Parallel backpropagation through each block

2) Cumulative sum (parallelizable)

General framework

In the paper, we extend the framework to a broader set of models:

- Arbitrary residual connection functions:

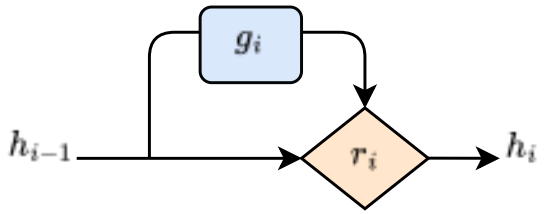


→ ResNets, post-layernorm transformers, …

General framework

In the paper, we extend the framework to a broader set of models:

- Arbitrary residual connection functions:



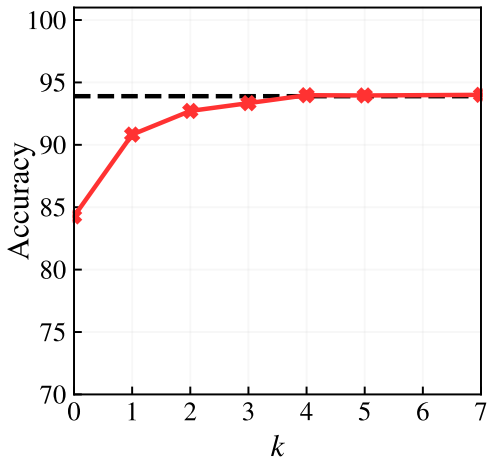
→ ResNets, post-layernorm transformers, …

- Handle intermediary losses:

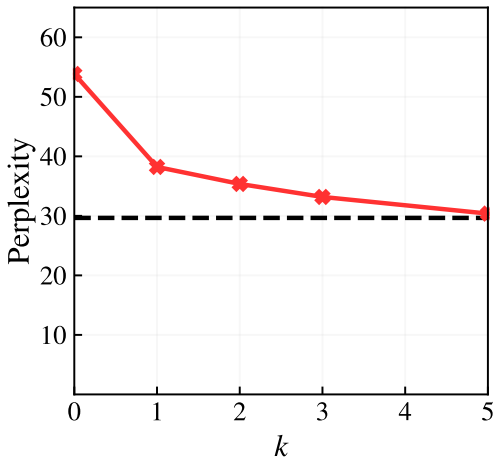
$$\mathcal{L}(x) = \sum_{i=0}^{L} \mathcal{L}_i(h_i)$$

→ RNNs: GRUs, LSTMs, …

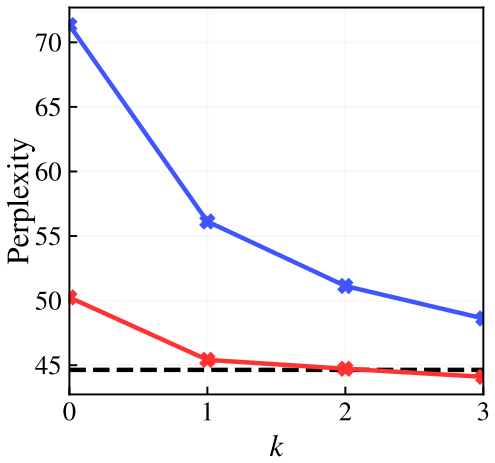
Result



ResNet110
CIFAR10
$(L = 54)$



Transformer
Wikitext103
$(L = 24)$



RNN (GRU x3)
Wikitext103
$(L = 256)$

→ Matches backpropagation with $k \ll L$ iterations.

Thank you!

Contact: erwan.fagnou@dauphine.psl.eu

Paper: