Gated Linear Attention Transformers with Hardware-Efficient Training

Songlin Yang*[1] Bailin Wang*[1] Yikang Shen[2] Rameswar Panda[2] Yoon Kim[1]

[1]Massachusetts Institute of Technology [2]MIT-IBM Watson AI Lab

Summary

Linear attention: removes the softmax in ordinary attention $\Longrightarrow$ a linear RNN with matrix-valued hidden states.

	Softmax Attention	Linear Attention
Training	$\mathbf{O} = \mathrm{softmax}\left((\mathbf{Q}\mathbf{K}^\top) \odot \mathbf{M}\right)\mathbf{V}$	$\mathbf{O} = \left((\mathbf{Q}\mathbf{K}^\top) \odot \mathbf{M}\right)\mathbf{V}$
Inference	$\boldsymbol{o}_t = \frac{\sum_{i=1}^t \exp(\boldsymbol{q}_t \boldsymbol{k}_i^\top)\boldsymbol{v}_i}{\sum_{i=1}^t \exp(\boldsymbol{q}_t \boldsymbol{k}_i^\top)}$	$\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{k}_t^\top \boldsymbol{v}_t,\ \boldsymbol{o}_t = \boldsymbol{q}_t \mathbf{S}_t$

Issues:

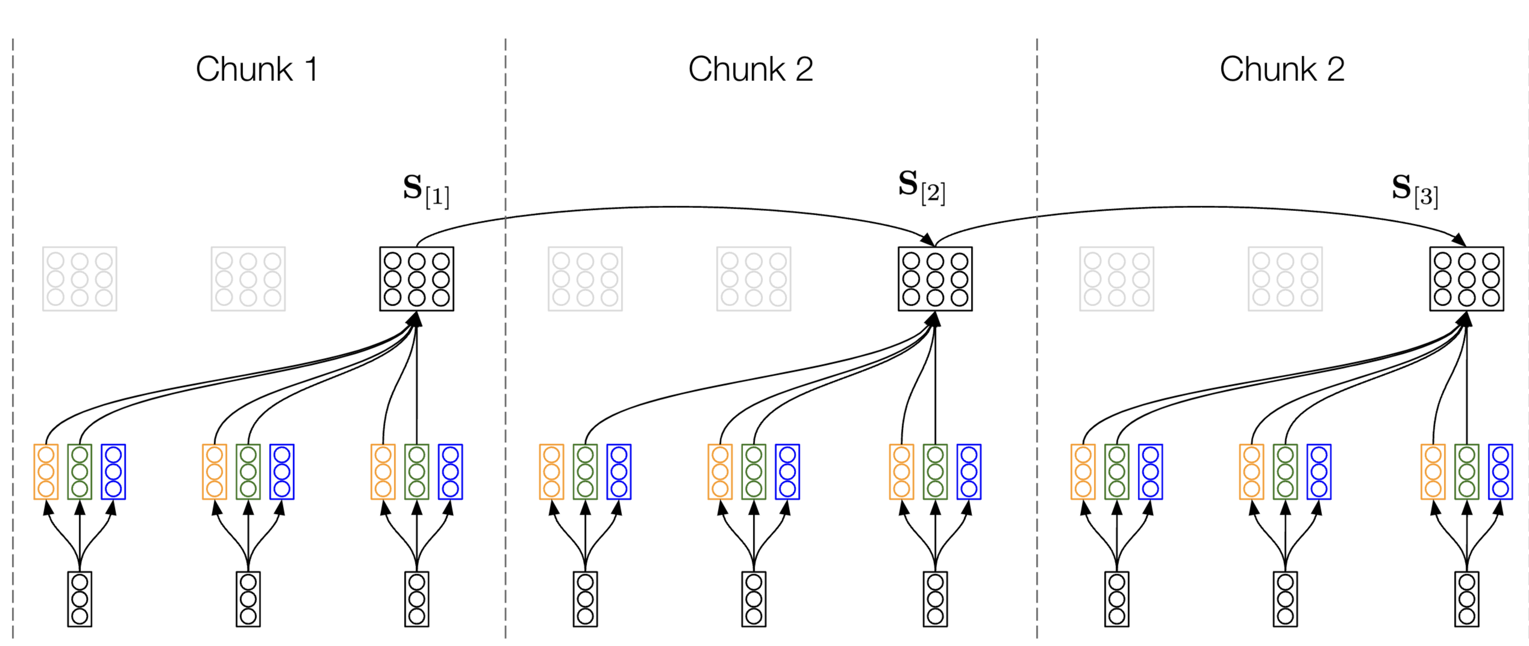
- Slow wall time training speed compared to FlashAttention.
- Poor language modeling performance.

Our contributions

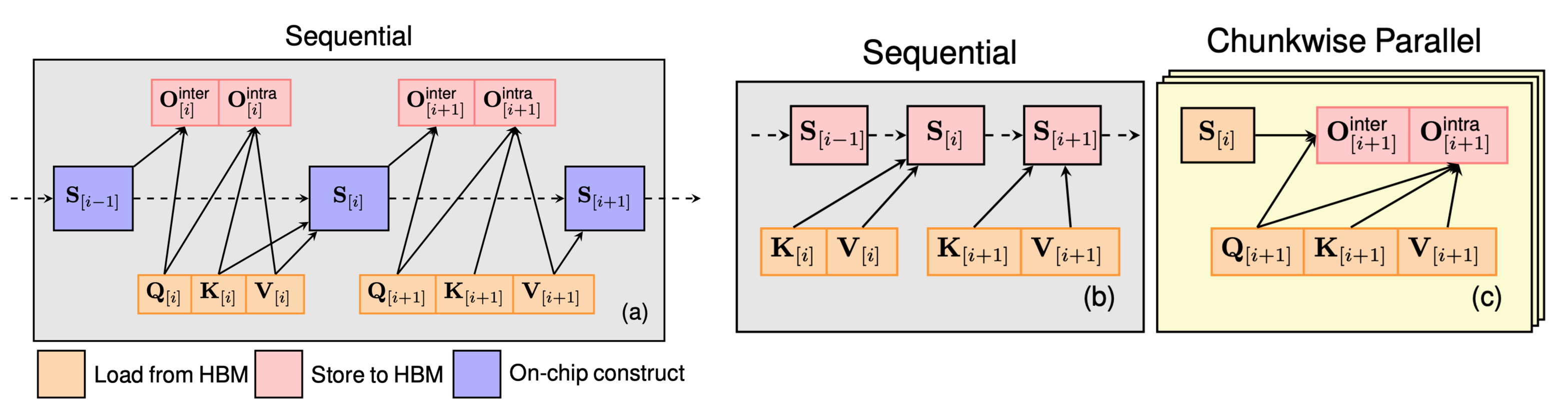
- **FlashLinearAttention**: a hardware-efficient linear attention implementation library.
- **Gated Linear Attention**: improve language modeling performance through a data-dependent gating mechanism.

Three Forms of Linear Attention

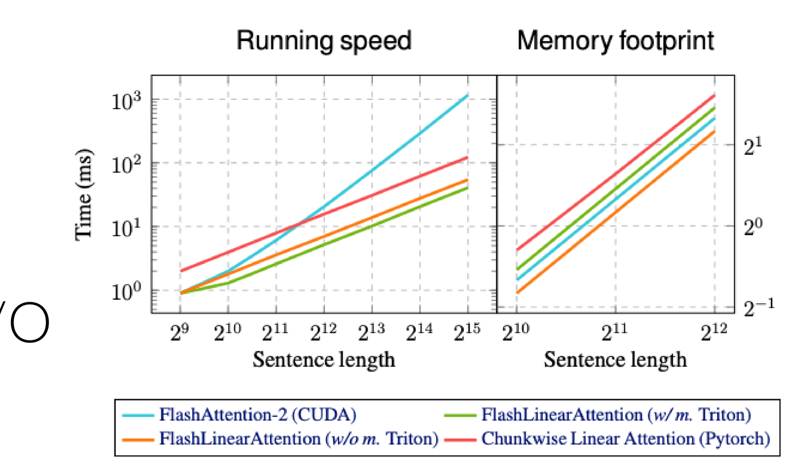
	Equation	Linear scaling	Tensor cores	Sequence parallel
Parallel	$\mathbf{O} = \left((\mathbf{Q}\mathbf{K}^\top) \odot \mathbf{M}\right)\mathbf{V}$	No, $O(L^2D)$	Yes	Yes
Recurrent	$\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{k}_t^\top \boldsymbol{v}_t$ $\boldsymbol{o}_t = \boldsymbol{q}_t \mathbf{S}_t$	Yes, $O(LD^2)$	No	No
Chunkwise	$\mathbf{S}_{[i+1]} = \mathbf{S}_{[i]} + \mathbf{K}_{[i]}^\top \mathbf{V}_{[i]}$ $\mathbf{O}_{[i+1]} = \mathbf{Q}_{[i+1]}\mathbf{S}_{[i]}$ $+\left((\mathbf{Q}_{[i+1]}\mathbf{K}_{[i+1]}^\top) \odot \mathbf{M}\right)\mathbf{V}_{[i+1]}$	Yes $O(LCD + LD^2)$	Yes,	Yes



FlashLinearAttention: Efficient Linear Attention



- (a): minimal I/O cost, restricted parallelism
- (b-c): high chunk-level parallelism, slightly higher I/O cost.



Gated Linear Attention

Introducing 2D forget gate $\mathbf{G}_t \in \mathbb{R}^{d \times d}$ to linear attention:

$$\mathbf{S}_t = \mathbf{G}_t \odot \mathbf{S}_{t-1} + \boldsymbol{k}_t^\top \boldsymbol{v}_t$$

Different parameterization on $\mathbf{G}_t$ leads to different models:

Model	Parameterization	Paramet
Mamba [Gu & Dao 2023]	$\mathbf{G}_t = \exp(-(\mathbf{1}\boldsymbol{\alpha}_t^\top) \odot \exp(\boldsymbol{A})),\ \ \boldsymbol{\alpha}_t = \mathrm{softplus}(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_1} \boldsymbol{W}_{\alpha_2})$	$\boldsymbol{A}, \boldsymbol{W}_{\alpha_1},$
Mamba-2 [Dao & Gu 2024]	$\mathbf{G}_t = \gamma_t \mathbf{1}\mathbf{1}^\top,\ \ \gamma_t = \exp\left(-\mathrm{softplus}\left(\boldsymbol{x}_t \boldsymbol{W}_\gamma\right)\exp\left(a\right)\right)$	$\boldsymbol{W}_\gamma, a$
xLSTM [Beck et al. 2024]	$\mathbf{G}_t = \gamma_t \mathbf{1}\mathbf{1}^\top,\ \ \gamma_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_\gamma\right)$	$\boldsymbol{W}_\gamma$
GLA [Yang et al. 2023]	$\mathbf{G}_t = \boldsymbol{\alpha}_t \mathbf{1}^\top,\ \ \boldsymbol{\alpha}_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_1} \boldsymbol{W}_{\alpha_2}\right)^{\frac{1}{\tau}}$	$\boldsymbol{W}_{\alpha_1}, \boldsymbol{W}$
Gated RetNet [Sun et al. 2024]	$\mathbf{G}_t = \gamma_t \mathbf{1}\mathbf{1}^\top,\ \ \gamma_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_\gamma\right)^{\frac{1}{\tau}}$	$\boldsymbol{W}_\gamma$
HGRN-2 [Qin et al. 2024]	$\mathbf{G}_t = \boldsymbol{\alpha}_t \mathbf{1}^\top,\ \ \boldsymbol{\alpha}_t = \boldsymbol{\gamma} + (1-\boldsymbol{\gamma})\sigma(\boldsymbol{x}_t \boldsymbol{W}_\alpha)$	$\boldsymbol{W}_\alpha, \boldsymbol{\gamma}$
RWKV-6 [Peng et al. 2024]	$\mathbf{G}_t = \boldsymbol{\alpha}_t \mathbf{1}^\top,\ \ \boldsymbol{\alpha}_t = \exp\left(-\exp\left(\boldsymbol{x}_t \boldsymbol{W}_\alpha\right)\right)$	$\boldsymbol{W}_\alpha$
Gated RFA [Peng et al. 2021]	$\mathbf{G}_t = \gamma_t \mathbf{1}\mathbf{1}^\top,\ \ \gamma_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_\gamma\right)$	$\boldsymbol{W}_\gamma$
Decaying FW [Mao et al. 2022]	$\mathbf{G}_t = \boldsymbol{\alpha}_t \boldsymbol{\beta}_t^\top,\ \ \boldsymbol{\alpha}_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_\alpha\right), \boldsymbol{\beta}_t = \sigma\left(\boldsymbol{x}_t \boldsymbol{W}_\beta\right)$	$\boldsymbol{W}_\gamma$

Gated Linear Attention $\subset$ State-Space Models

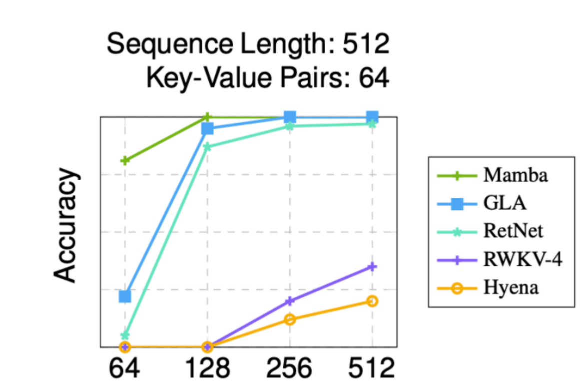
GLA's chunkwise parallel form and fast Triton kernel:

- Support efficient scaling of hidden state size by leveraging tensor cores.
- Faciliate training of recent models like **HGRN-2**, **RWKV-6**, **Mamba-2**.

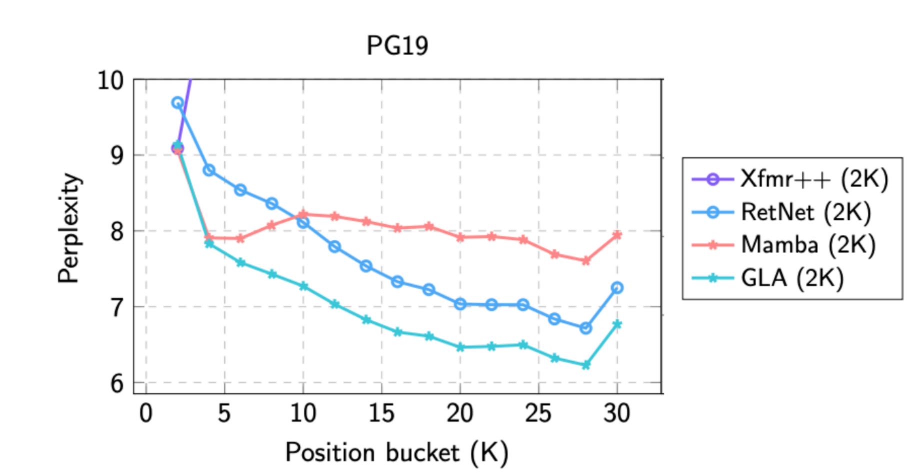
Performance

Scale	Model	Wiki. ppl ↓	LM Eval. acc. ↑	Recall Tasks FDA	SWD	SQD
340M Params 15B Tokens	Transformer++	28.39	41.2	21.4	42.2	22.1
	RetNet	32.33	41.0	2.9	13.3	27.6
	Mamba	28.39	41.8	2.1	12.4	23.0
	GLA	28.65	41.5	8.1	18.6	27.2
1.3B Params 100B Tokens	Transformer++	16.85	50.9	21.4	42.2	22.1
	RetNet	18.64	48.9	14.3	42.8	34.7
	Mamba	17.06	50.0	6.2	41.4	35.2
	GLA	17.22	51.0	19.9	50.6	42.6

MQAR



Length extrapolation



Training Speed / Memory