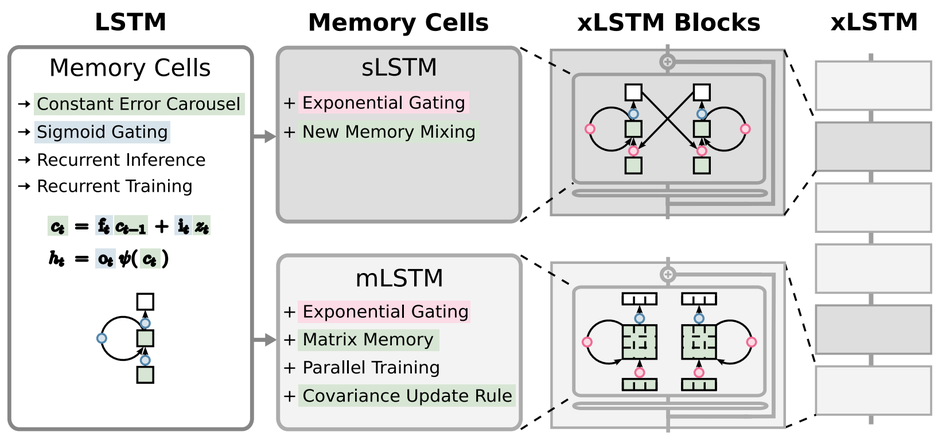
xLSTM
Extended Long Short-Term Memory

Maximilian Beck*[1,2], Korbinian Pöppel*[1,2], Markus Spanring[1], Andreas Auer[1,2], Oleksandra Prudnikova[1], Michael Kopp, Günter Klambauer[1,2], Johannes Brandstetter[1,2,3], Sepp Hochreiter[1,2,3]

*Equal Contribution

TL;DR

We extend the LSTM by Exponential Gating and Matrix Memory and outperform Transformers and State Space Models on Language Modeling.



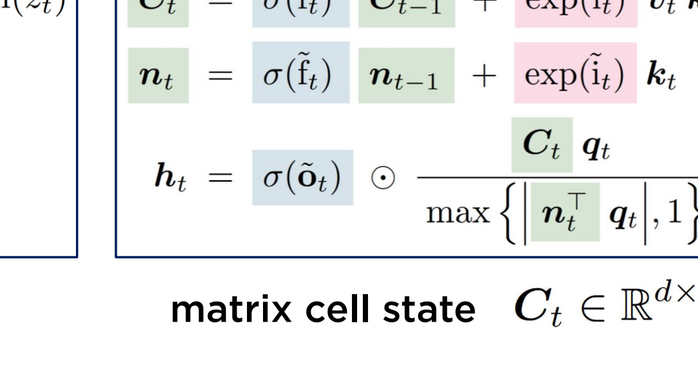
Review of the LSTM

- The LSTM with its constant error carousel and gating was introduced to overcome the vanishing gradient problem of RNNs:

$$
\begin{aligned}
c_t &= f_t\; c_{t-1} + i_t\; z_t && \text{cell state}\\
h_t &= o_t\; \tanh\left(c_t\right) && \text{hidden state}\\
z_t &= \tanh\left(\tilde{z}_t\right)\,, && \text{cell input}\\
i_t &= \sigma\left(\tilde{i}_t\right)\,, && \text{input gate}\\
f_t &= \sigma\left(\tilde{f}_t\right)\,, && \text{forget gate}\\
o_t &= \sigma\left(\tilde{o}_t\right)\,, && \text{output gate}\\
g_t &= \boldsymbol{w}_g^\top\, \boldsymbol{x}_t + r_g\, h_{t-1} + b_g\,, && \text{pre-activations}\\
g &= \{\tilde{i},\, \tilde{f},\, \tilde{o},\, \tilde{z}\} && \text{gates}
\end{aligned}
$$

LSTM Limitations

- Inability to revise storage decisions
- Limited storage capacity
- Efficiency: Lack of parallelizability



Left: Nearest Neighbor Search problem (MSE). *Right:* Rare Token Prediction. Perplexity (PPL) of token prediction on Wikitext-103 in partitions of token frequency.

Extended Long Short-Term Memory

- To overcome the LSTM limitations, xLSTM introduces **two main modifications** to the LSTM: **Exponential Gating & Matrix Memory**
- Both enrich the LSTM family by two new members:
  i. the **sLSTM** with a scalar memory, a scalar update and memory mixing
  ii. the **mLSTM** with a matrix memory and a covariance (outer product) update rule
- The xLSTM architecture contains a stack of sLSTM and mLSTM blocks in the pre-LayerNorm residual backbone.
- We use the notation xLSTM[a:b] for the ratio a/b of mLSTM vs. sLSTM blocks

Exponential Gating

sLSTM

$$
\begin{aligned}
c_t &= \sigma(\tilde{f}_t)\; c_{t-1} + \exp(\tilde{i}_t)\; \tanh(\tilde{z}_t)\\
n_t &= \sigma(\tilde{f}_t)\; n_{t-1} + \exp(\tilde{i}_t)\\
h_t &= \sigma(\tilde{o}_t)\; \frac{c_t}{n_t}
\end{aligned}
$$

scalar cell state $c_t \in \mathbb{R}$

mLSTM

$$
\begin{aligned}
\boldsymbol{C}_t &= \sigma(\tilde{f}_t)\; \boldsymbol{C}_{t-1} + \exp(\tilde{i}_t)\; \boldsymbol{v}_t\, \boldsymbol{k}_t^\top\\
\boldsymbol{n}_t &= \sigma(\tilde{f}_t)\; \boldsymbol{n}_{t-1} + \exp(\tilde{i}_t)\; \boldsymbol{k}_t\\
\boldsymbol{h}_t &= \sigma(\tilde{\boldsymbol{o}}_t) \odot \frac{\boldsymbol{C}_t\, \boldsymbol{q}_t}{\max\left\{\left|\boldsymbol{n}_t^\top\, \boldsymbol{q}_t\right|, 1\right\}}
\end{aligned}
$$

matrix cell state $\boldsymbol{C}_t \in \mathbb{R}^{d\times d}$

Gate/Input pre-activations:

$$
\begin{aligned}
g_t &= \boldsymbol{w}_g^\top\, \boldsymbol{x}_t + r_g\, h_{t-1} + b_g\\
g &= \{\tilde{i},\, \tilde{f},\, \tilde{o},\, \tilde{z}\}
\end{aligned}
$$

New Memory Mixing

$$
\begin{aligned}
\{\boldsymbol{q}_t, \boldsymbol{k}_t, \boldsymbol{v}_t\} &= \boldsymbol{W}_{\{q,k,v\}}\, \boldsymbol{x}_t + \boldsymbol{b}_{\{q,k,v\}}\\
\{\tilde{i}_t, \tilde{f}_t\} &= \boldsymbol{w}_{\{i,f\}}^\top\, \boldsymbol{x}_t + b_{\{i,f\}}\\
\tilde{\boldsymbol{o}}_t &= \boldsymbol{W}_{\boldsymbol{o}}\, \boldsymbol{x}_t + \boldsymbol{b}_{\boldsymbol{o}}
\end{aligned}
$$

Parallel Training

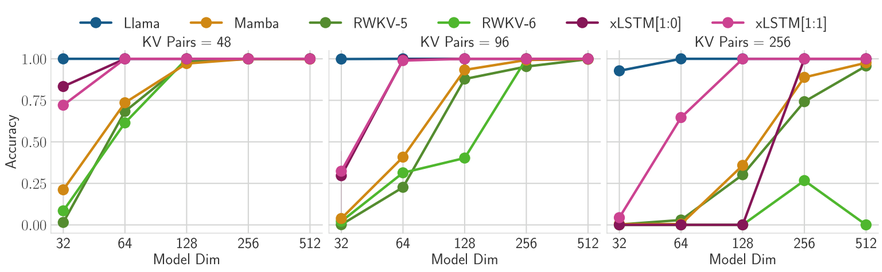
Results on Synthetic Tasks

- xLSTM with sLSTM (memory mixing) can solve formal language tasks, i.e. has state tracking capabilities

	Context Sensitive		Deterministic Context Free		Regular					
	Bucket Sort	Missing Duplicate	Mod Arithmetic (w Brackets)	Solve Equation	Cycle Nav	Even Pairs	Mod Arithmetic (w/o Brackets)	Parity	Majority	Majority Count
Llama	0.92 ± 0.02	0.08 ± 0.0	0.02 ± 0.0	0.02 ± 0.0	0.04 ± 0.01	1.0 ± 0.0	0.03 ± 0.0	0.03 ± 0.01	0.37 ± 0.01	0.13 ± 0.0
Mamba	0.69 ± 0.0	0.15 ± 0.0	0.04 ± 0.01	0.05 ± 0.02	0.86 ± 0.04	1.0 ± 0.0	0.05 ± 0.02	0.13 ± 0.02	0.69 ± 0.01	0.45 ± 0.03
RWKV-6	0.96 ± 0.0	0.23 ± 0.06	0.09 ± 0.01	0.09 ± 0.02	0.31 ± 0.14	1.0 ± 0.0	0.16 ± 0.0	0.22 ± 0.12	0.76 ± 0.01	0.24 ± 0.01
LSTM	0.94 ± 0.01	0.2 ± 0.0	0.72 ± 0.04	0.38 ± 0.05	0.93 ± 0.07	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.82 ± 0.02	0.33 ± 0.0
xLSTM[0:1]	0.84 ± 0.08	0.23 ± 0.01	0.57 ± 0.09	0.55 ± 0.09	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0	0.75 ± 0.02	0.22 ± 0.0
xLSTM[1:1]	0.7 ± 0.21	0.2 ± 0.01	0.15 ± 0.06	0.24 ± 0.04	0.8 ± 0.03	1.0 ± 0.0	0.6 ± 0.4	1.0 ± 0.0	0.64 ± 0.04	0.5 ± 0.0

Formal language tasks results given by the scaled accuracy of different models, grouped by the Chomsky hierarchy.

- xLSTM with mLSTM (matrix memory) performs best on extended version of the MQAR task.



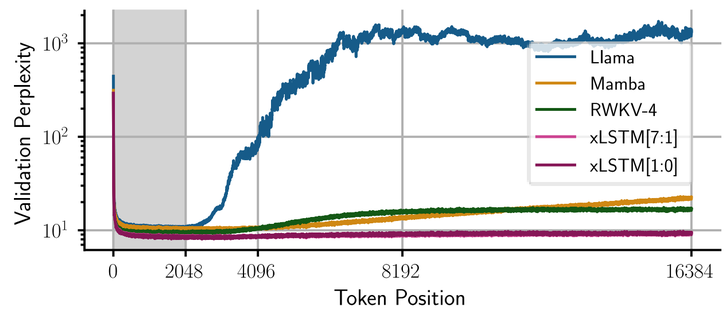
Multi-Query Associative Recall (MQAR) task with context length 2048 for different numbers of key-value pairs. The x-axis displays the model size and the y-axis the validation accuracy.

Results on Language Modeling

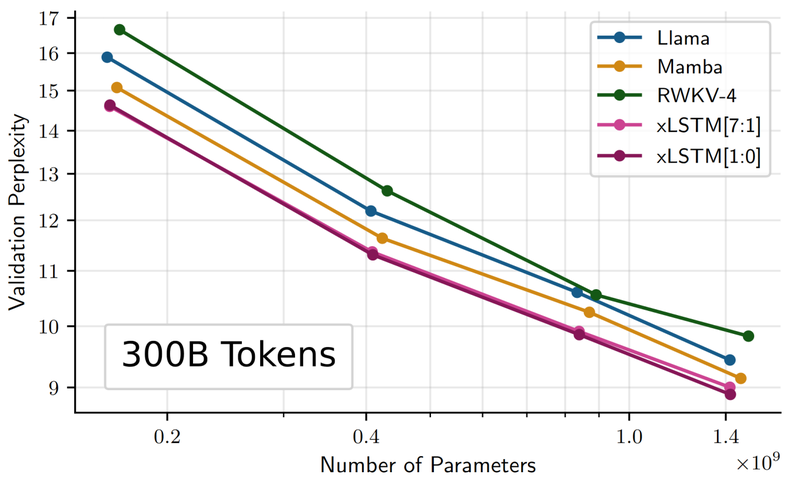
- Training on 15B / 300B tokens from the SlimPajama Dataset
- All models use the GPT2 tokenizer
- Model sizes: 125M, 350M, 760M, 1.3B
- Context length: 2048

Model	#Params M	SlimPajama (15B) ppl ↓
GPT-3	356	14.26
Llama	407	14.25
H3	420	18.23
Mamba	423	13.70
Hyena	435	17.59
RWKV-4	430	15.62
RWKV-5	456	14.25
RWKV-6	442	15.03
RetNet	431	16.23
HGRN	411	17.59
GLA	412	16.15
HGRN2	411	14.32
xLSTM[1:0]	409	**13.43**
xLSTM[7:1]	408	13.48

Validation Perplexities (ppl) on SlimPajama (15B). All models are trained on 15B tokens of SlimPajama with context length 2048.



Comparison of 1.3B-sized models trained on 300B tokens with context length 2048 and then tested for context lengths up to 16384.



Next token prediction PPL on validation set when trained on 300B tokens from SlimPajama.

beck@ml.jku.at, poeppel@ml.jku.at
maxmbeck, KorbiPoeppel
Paper: arxiv.org/abs/2405.04517
Video: https://youtu.be/KuRpxvMMrlk?si=sw9vJwIJyxXyrp7r
Code: github.com/NX-AI/xlstm

1 ELLIS Unit Linz and LIT AI Lab, Institute for Machine Learning, Johannes Kepler University, Linz, Austria
2 NXAI Lab, Linz, Austria
3 NXAI GmbH, Linz, Austria