xLSTM: Recurrent Neural Network Architectures for Scalable and Efficient Large Language Models

Maximilian Beck
Institute for Machine Learning
Johannes Kepler University Linz, Austria
beck@ml.jku.at – maxbeck.ai

1 Motivation

The widespread adoption of Large Language Models (LLMs) is transforming fields ranging from healthcare [12, 20] and education [35] to software development [20] and customer service, while also changing how we search for information, communicate, and manage our everyday tasks. As LLMs are applied to more and more areas, the computational and energy demands on both training and inference continue to grow [8]. Training requires ever greater computational power as context lengths grow, models scale up, and training dataset size rises, while inference must remain fast and affordable across cloud platforms and edge devices. These challenges highlight the need for architectures that can deliver efficiency without sacrificing performance, even under resource constraints.

The backbone of most of todays LLMs is the Transformer architecture which processes and generates text as a sequence of tokens, where each token consists of one or several characters [34]. At its core the Transformer architecture relies on the self-attention mechanism, which captures long-range dependencies and contextual relationships by computing pairwise interactions between all tokens in the sequence. This leads to quadratic scaling in compute and linear scaling in memory with respect to the sequence length, as all tokens must be stored in memory for the attention computation. In scenarios with very long sequences or resource constraints the self-attention mechanism creates significant challenges.

In contrast, traditional recurrent neural networks (RNNs), such as the Long Short-Term Memory (LSTM) [13, 15, 16] process sequences in a step-by-step manner and thus exhibit only linear complexity with respect to the sequence length. LSTMs control the error and information flow using a gating mechanism that consists of input, forget and output gates. In combination, these gates maintain and update a fixed memory state at each time step, effectively compressing the history of the past inputs into a constant-size memory. LSTMs have been successfully applied in various domains [17, 14, 28], are still widely used in highly relevant applications [23], and have been even used for early language models [21]. However, due to their sequential nature, traditional RNNs are inherently less parallelizable and hence less efficient during training compared to Transformers, which can process all tokens simultaneously. This limited the scalability of early LSTM-based language models and paved the way for the emergence of the Transformer architecture in language modeling.

My thesis aims to overcome the challenges of current Transformer architectures by developing new recurrent neural network architectures that combine the benefits of both – RNNs' linear scaling with sequence length and low memory requirements, with Transformers' parallel training and strong performance on natural language tasks. In **[a]**, we introduce the xLSTM a new RNN architecture with exponential gating and a new scalar and matrix memory structure and show that the xLSTM performs favorably when compared to state-of-the-art Transformers and State Space Models, both in performance and scaling. In **[b]**, we develop Tiled Flash Linear Attention (TFLA) a new efficient hardware-aware kernel algorithm for the xLSTM with matrix memory, enabling large scale training of xLSTM models. In **[c]**, we introduce a 7 billion parameter xLSTM LLM that leverages the fast xLSTM TFLA kernels and combines xLSTM's architectural benefits with targeted optimizations for fast end efficient inference.

2 Approaches & Results

This thesis is dedicated to establish RNNs as a competitive alternative to the Transformer architecture for LLMs, in order to overcome the computational and memory challenges posed by the quadratic self-attention mechanism. We approach this goal from three angles: **[a]** Novel RNN architectures that combine the strengths of LSTMs and Transformers, **[b]** Hardware-aware algorithms and implementations for efficient training of large scale RNNs, and **[c]** Scaling RNN models to billions of parameters and optimizing them for fast and efficient inference.

xLSTM Architecture. Despite their tremendous successes, the original LSTM has three main limitations: (1) Inability to revise storage decisions, (2) Limited storage capacity, (3) Lack of parallelizability, which results in training inefficiencies at large scale. To overcome these limitations, the Extended Long Short-Term Memory (xLSTM) introduces two key innovations: (i) Exponential Gating, which allows the model to revise its storage decisions over time, and (ii) a dual memory system consisting of the sLSTM, a scalar memory cell, with scalar update rule and memory mixing – and the mLSTM, a matrix memory, with outer product update rule and full parallelizability. By integrating these memory cells into residual blocks, and stacking multiple such blocks, we obtain the xLSTM architecture. In **[a]**, we demonstrate the effectiveness of the xLSTM architecture on synthetic benchmarks, as well as on real-world language modeling tasks, where it outperforms comparable Transformers architectures [32], State Space Models [11] and other RNNs [36, 31, 24, 25, 27] in terms of both performance and scaling.

Hardware-aware Algorithms & Implementations for Linear RNNs. Even though RNNs' linear scaling in compute and constant memory requirements of RNNs such as the xLSTM offer theoretical advantages over Transformers, realizing these benefits in practice requires hardware optimized algorithms and implementations, because Transformers rely on the highly optimized Flash Attention kernels [6, 5, 30]. To overcome this issue, we introduce Tiled Flash Linear Attention (TFLA) in **[b]**, a new algorithm for efficient training of linear RNNs such as the mLSTM. TFLA combines chunkwise parallel training [36, 37] with a novel tiling stategy of the matrix computations in sequence dimension in order to fully leverage the memory hierarchy of modern hardware. Our benchmarks show that our mLSTM kernels based on TFLA achieve significant improvements in memory consumption and runtime compared to previous linear RNN or highly optimized Flash Attention kernels.

Scaling up xLSTM models. So far in **[a]**, xLSTM models have been scaled up to 1.4B parameters. However, many widely used open-source LLM models use at least 7B parameters [32, 33, 10, 19]. Therefore, in order to demonstrate the competitive performance of xLSTM models at scale, we scale up the xLSTM architecture to 7B parameters in **[c]**. For this, we leverage the fast TFLA mLSTM kernels from **[b]**, and further optimize the xLSTM architecture for high training efficiency and stability, as well as for fast and efficient inference. Specifically, the new xLSTM 7B architecture fully relies on mLSTM cells with parallel training mode and placed in optimized mLSTM blocks. The optimizations include adapting the mLSTM memory dimension and (re-)adding position-wise feedforward MLP layers. We find that the resulting xLSTM 7B architecture with the modified block design achieves a $2\times$ to $4\times$ higher token throughput compared to the previous xLSTM block design. In our evaluations on language downstream and long context tasks, xLSTM 7B shows comparable performance to Transformers and Mamba models of the same size, while achieving the highest prefill and generation throughput with the lowest GPU memory footprint on our inference efficiency benchmarks.

2.1 Future Work

Scaling Laws of xLSTM models. Scaling laws play a central role in guiding the development of LLMs, by providing guidance on model parameter and dataset size allocation, as well as architecture design choices – so far mainly for Transformer based architectures [22, 18]. In **[d]**, we are currently investigating the scaling properties of xLSTM across several dimensions and orders of magnitude, aiming to provide a rigourous empirical foundation for future xLSTM model development as well as insights into the differences between xLSTM and Transformer scaling.

3 Contributions

The following list presents the papers that I authored either as single first or shared first author and that will form the **main part of my thesis**:

[a] xLSTM: Extended Long Short-Term Memory. *NeurIPS, 2024* [4].

[b] Tiled Flash Linear Attention: More Efficient Linear RNN and xLSTM Kernels. *Under Review, Arxiv, 2025* [3].

[c] xLSTM 7B: A Recurrent LLM for Fast and Efficient Inference. *International Conference on Machine Learning (ICML), 2025* [2].

Work in progress:

[d] xLSTM Scaling Laws: Competitive Performance with Linear Time-Complexity

The following list presents **additional papers** that I (co-)authored, including my contributions:

1. FlashRNN: Optimizing Traditional RNNs on Modern Hardware. *International Conference on Learning Representations (ICLR), 2025*. [26]
   Contribution: I implemented the FlashRNN sLSTM and LSTM Triton kernels. I set up and conducted the speed experiments and presented the results in the paper.
2. Vision-LSTM: xLSTM as Generic Vision Backbone. *International Conference on Learning Representations (ICLR), 2025*. [1]
   Contribution: I provided the core mLSTM implementation.
3. A Large Recurrent Action Model: xLSTM enables Fast Inference for Robotics Tasks. *International Conference on Machine Learning (ICML), 2025*. [29]
   Contribution: I contributed the fast Triton inference kernels, and provided feedback on the manuscript.
4. Addressing Parameter Choice Issues in Unsupervised Domain Adaptation by Aggregation. *International Conference on Learning Representations (ICLR), 2023*. [7]
   Contribution: I developed the implementation and source code of our method IWA as well the baselines. I conducted experiments and presented the results in the paper.
5. Few-Shot Learning by Dimensionality Reduction in Gradient Space. *Conference on Lifelong Learning Agents (CoLLAs), 2022*. [9]
   Contribution: I conducted experiments and provided baseline implementations. I contributed to the writing of the experiments section.

Impact. To date, my research has been cited 644 times as reported by Google Scholar.[1] The open source xLSTM GitHub repository of which I am core contributor has more than 2k stars and the mLSTM TFLA Triton kernels repository has 68 stars.[2]. The xLSTM architecture and technology is being developed further by NX-AI, a spin-off company of JKU Linz.

Invited Talks. Besides internal presentations at the Institute for Machine Learning at JKU Linz, I have been invited 13 times during my PhD to present my research at universities (FAU Erlangen-Nuremberg, KIT Karlsruhe ALR, Ruhr-University Bochum), industry research labs (Meta, Google Research, ISTA, G-Research), and industry conferences (Machine Learning Week Europe).[3]

Miscellaneous:

- Part of the **ELLIS PhD Program**
- Internship at **Meta FAIR** in Paris from May to October 2025
- Reviewer for NeurIPS-2024, ICLR-2025, ICML-2025, NeurIPS-2025

[1] Google Scholar [2] xLSTM repo, TFLA repo [3] Full List of Talks

References

[1] Benedikt Alkin, Maximilian Beck, Korbinian Pöppel, Sepp Hochreiter, and Johannes Brandstetter. Vision-LSTM: xLSTM as generic vision backbone. In *The Thirteenth International Conference on Learning Representations*, 2025.

[2] Maximilian Beck, Korbinian Pöppel, Phillip Lippe, Richard Kurle, Patrick M Blies, Günter Klambauer, Sebastian Böck, and Sepp Hochreiter. xLSTM 7b: A recurrent LLM for fast and efficient inference. In *Forty-second International Conference on Machine Learning*, 2025.

[3] Maximilian Beck, Korbinian Pöppel, Phillip Lippe, and Sepp Hochreiter. Tiled Flash Linear Attention: More efficient linear RNN and xLSTM kernels, 2025.

[4] Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xLSTM: Extended long short-term memory. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024.

[5] T. Dao. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. In *The Twelfth International Conference on Learning Representations*, 2024.

[6] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[7] Marius-Constantin Dinu, Markus Holzleitner, Maximilian Beck, Hoan Duc Nguyen, Andrea Huber, Hamid Eghbal-zadeh, Bernhard A. Moser, Sergei Pereverzyev, Sepp Hochreiter, and Werner Zellinger. Addressing parameter choice issues in unsupervised domain adaptation by aggregation. In *The Eleventh International Conference on Learning Representations*, 2023.

[8] Cooper Elsworth, Keguo Huang, David Patterson, Ian Schneider, Robert Sedivy, Savannah Goodman, Ben Townsend, Parthasarathy Ranganathan, Jeff Dean, Amin Vahdat, Ben Gomes, and James Manyika. Measuring the environmental impact of delivering ai at google scale, 2025.

[9] Martin Gauch, Maximilian Beck, Thomas Adler, Dmytro Kotsur, Stefan Fiel, Hamid Eghbalzadeh, Johannes Brandstetter, Johannes Kofler, Markus Holzleitner, Werner Zellinger, Daniel Klotz, Sepp Hochreiter, and Sebastian Lehner. Few-shot learning by dimensionality reduction in gradient space. In Sarath Chandar, Razvan Pascanu, and Doina Precup, editors, *The Conference on Lifelong Learning Agents*, 2022.

[10] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, et al. The llama 3 herd of models, 2024.

[11] A. Gu and T. Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *International Conference on Learning Representations*, 2024.

[12] Kai He, Rui Mao, Qika Lin, Yucheng Ruan, Xiang Lan, Mengling Feng, and Erik Cambria. A survey of large language models for healthcare: from data, technology, and applications to accountability and ethics, 2025.

[13] S. Hochreiter. Untersuchungen zu dynamischen neuronalen Netzen. Master's thesis, Technische Universität München, 1991.

[14] S. Hochreiter, M. Heusel, and K. Obermayer. Fast model-based protein homology detection without alignment. *Bioinformatics*, 23(14):1728–1736, 2007.

[15] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[16] S. Hochreiter and J. Schmidhuber. LSTM can solve hard long time lag problems. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 9, pages 473–479. MIT Press, Cambridge MA, 1997.

[17] S. Hochreiter, A. Steven Younger, and Peter R. Conwell. Learning to learn using gradient descent. In G. Dorffner, H. Bischof, and K. Hornik, editors, *Proc. Int. Conf. on Artificial Neural Networks (ICANN 2001)*, pages 87–94. Springer, 2001.

[18] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022.

[19] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[20] Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. A survey on large language models for code generation, 2024.

[21] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling, 2016.

[22] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

[23] G. Nearing, D. Cohen, V. Dube, M. Gauch, O. Gilon, S. Harrigan, A. Hassidim, D. Klotz, F. Kratzert, A. Metzger, S. Nevo, F. Pappenberger, C. Prudhomme, G. Shalev, S. Shenzis, T. Y. Tekalign, D. Weitzner, and Y. M. B. Kosko. Global prediction of extreme floods in ungauged watersheds. *Nature*, 627:559–563, 2024.

[24] B. Peng, E. Alcaide, Q. Anthony, A. Albalak, S. Arcadinho, H. Cao, X. Cheng, M. Chung, M. Grella, K. K. GV, X. He, H. Hou, P. Kazienko, J. Kocon, J. Kong, B. Koptyra, H. Lau, K. S. I. Mantri, F. Mom, A. Saito, X. Tang, B. Wang, J. S. Wind, S. Wozniak, R. Zhang, Z. Zhang, Q. Zhao, P. Zhou, J. Zhu, and R.-J. Zhu. RWKV: Reinventing RNNs for the transformer era. *ArXiv*, 2305.13048, 2023.

[25] Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Xingjian Du, Teddy Ferdinan, Haowen Hou, Przemysław Kazienko, Kranthi Kiran GV, Jan Kocoń, Bartłomiej Koptyra, Satyapriya Krishna, Ronald McClelland Jr., Jiaju Lin, Niklas Muennighoff, Fares Obeid, Atsushi Saito, Guangyu Song, Haoqin Tu, Cahya Wirawan, Stanisław Woźniak, Ruichong Zhang, Bingchen Zhao, Qihang Zhao, Peng Zhou, Jian Zhu, and Rui-Jie Zhu. Eagle and finch: Rwkv with matrix-valued states and dynamic recurrence. *arXiv*, 2404.05892, 2024.

[26] Korbinian Pöppel, Maximilian Beck, and Sepp Hochreiter. FlashRNN: I/o-aware optimization of traditional RNNs on modern hardware. In *The Thirteenth International Conference on Learning Representations*, 2025.

[27] Zhen Qin, Songlin Yang, Weixuan Sun, Xuyang Shen, Dong Li, Weigao Sun, and Yiran Zhong. HGRN2: Gated linear RNNs with state expansion. In *First Conference on Language Modeling*, 2024.

[28] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015.

[29] Thomas Schmied, Thomas Adler, Vihang Prakash Patil, Maximilian Beck, Korbinian Pöppel, Johannes Brandstetter, Günter Klambauer, Razvan Pascanu, and Sepp Hochreiter. A large recurrent action model: xLSTM enables fast inference for robotics tasks, 2025.

[30] J. Shah, G. Bikshandi, Y. Zhang, V. Thakkar, P. Ramani, and T. Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. 2024.

[31] Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei. Retentive network: A successor to transformer for large language models. *ArXiv*, 2307.08621, 2023.

[32] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. `https://arxiv.org/abs/2302.13971`, 2023.

[33] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv*, 2307.09288, 2023.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[35] Shen Wang, Tianlong Xu, Hang Li, Chaoli Zhang, Joleen Liang, Jiliang Tang, Philip S. Yu, and Qingsong Wen. Large language models for education: A survey and outlook, 2024.

[36] Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *Forty-first International Conference on Machine Learning*, 2024.

[37] Songlin Yang and Yu Zhang. FLA: A Triton-based library for hardware-efficient implementations of linear attention mechanism. January 2024.