Is Attention all you need? The Search for a New Architecture

Michael Staniek

Department of Computational Linguistics
Heidelberg University

October 13, 2025

Outline

Grading

- 40% Project
- 30% Presentation
- 30% Participation

Presentation

- Students are expected to present a paper
- Paper list in this presentation
- Send email to me with two bold and (two not-bold preferences OR own paper ideas not yet included but could fit) until Thursday 23:59.
- staniek@cl.uni-heidelberg.de
- No lecture next week
- First two presenters (in 2 weeks) get a fixed slot in this seminar, need to decide today (easy papers)
  - xLSTM and RWKV are the first two papers.
- Next two presenters (in 3 weeks) can also be fixed today.
  - RetNet and S4
- Following week (with normal registration): Mamba, Mamba2

Further Schedule, Tentative

- Mamba, Mamba2
- Hyena, RoFormer
- Titans, Atlas
- Performer, Linforer...
- Rest: TBD

Grading

- https://www.uni-heidelberg.de/md/neuphil/gs/sprache02/hinweise/kriterienraster_referate.pdf
- Very important: Engaging the audience, not only looking at the computer/flash cards
- No mistakes in the presentation (typos, formulas)
- Good overview about paper and if available, criticism
- Good english skills (Niveau B2)

Participation

- Either: Presenters prepare 2 questions that everyone has to answer per mail
- Or: Students ask questions related to papers
- Grading: All questions asked/answered and participation during lectures/discussion

Project

- As usual, do a project at the end of the semester
- You can start now if you already have ideas
- Submission: Code+Short Report+Declaration of Authorship/Independend Work (german)
- Deadline: 31.03.2026 23:59:59:99999999999
- Grading: Implementation (Code submission) + Short Report
- If two people want to do a project together: Github/Gitlab usage from the beginning (no push at the end as "initial commit", thats only allowed for solo projects)

Grading

- 40% Project
- 30% Presentation
- 30% Participation

Motivation

- **Transformers** revolutionized NLP — first introduced in *"Attention is All You Need"* (Vaswani et al., 2017).
- Enabled **parallel training**, deep architectures, and superior results across tasks.
- However:
  - Inference remains **inefficient** — no hidden state summarizing previous context.
  - Context windows are **finite** and **expensive** to scale.
- $\rightarrow$ The community is actively searching for **new architectures** beyond Transformers.

Core Challenges

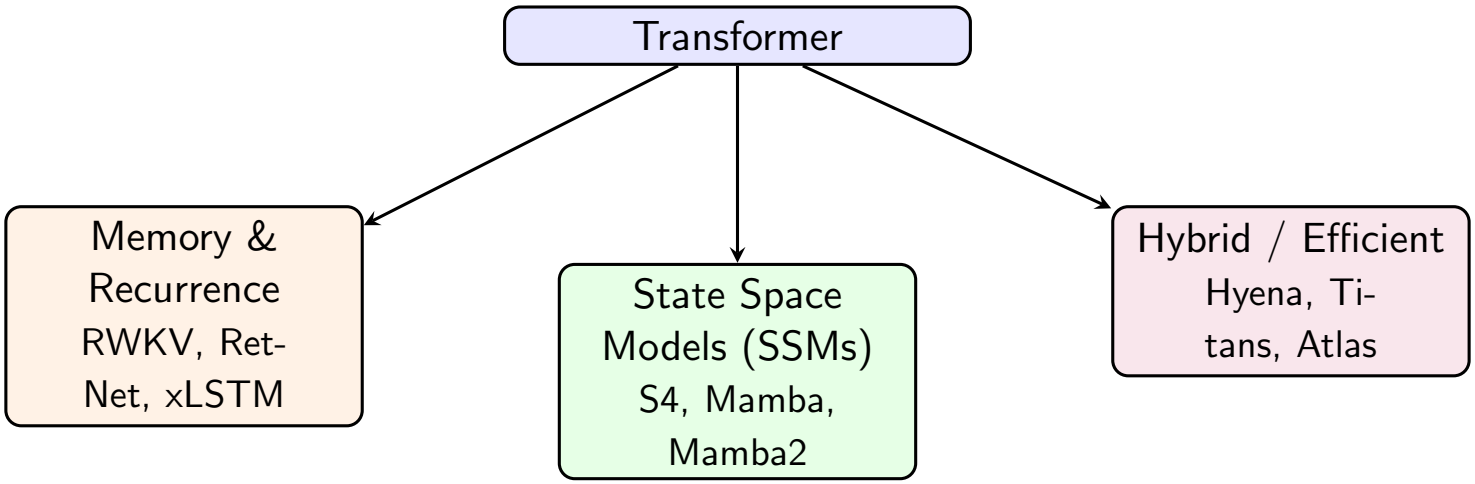
Training Efficiency

- Parallelization via attention.
- Long sequences: quadratic cost $O(n^2)$.
- Memory limits in large context training.

Inference Efficiency

- Autoregressive decoding = sequential.
- No persistent hidden state.
- KV cache helps, but scales linearly with context.

The Search for a New Architecture



Memory-Enhanced & Recurrence-Based Models

- **RWKV** [14]
- **RetNet** [19]
- **xLSTM** [1]
- TTT-MLP [18]

Memory-Enhanced & Recurrence-Based Models

- **RWKV** [14]
- **RetNet** [19]
- **xLSTM** [1]
- TTT-MLP [18]

Key Idea

Reintroduce **recurrence and statefulness** — efficient inference and persistent memory.

State Space Models (SSMs)

Goal: Efficient long-sequence modeling with linear time complexity.

- **S4** [11]
- **Mamba** [10]
- **Mamba-2** [9]

Advantages:

- Linear scaling ($O(n)$).
- Captures long-term dependencies (maybe?)
- Great potential for efficient inference.

Hybrid and Alternative Architectures

- **Hyena** [15]
- **Titans** [4]
- **Atlas** [2]
- **Miras** [3]

Trend

Mixing multiple paradigms: attention, recurrence, retrieval, and structure.

Sparse & Efficient Attention Models

- **RoFormer** [17]
- **Performer** [6]
- **Linformer** [22]
- **Reformer** [12]
- **Longformer** [5]
- **FNet** [13]
- Linear Transformers [21]
- Synthesizer [20]
- Flash Attention [8]
- Flash Attention 2 [7]
- Gated Linear Attention [23]
- BigBird [24]
- Compressive Transformers [16]

Goal: Retain Transformer-like behavior with reduced $O(n^2)$ bottleneck.

Other ideas

- Switch Transformer
- Diffusion-LM

Questions?

[1] 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, 2024.

[2] Ali Behrouz, Zeman Li, Praneeth Kacham, Majid Daliri, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. Atlas: Learning to optimally memorize the context at test time, 2025.

[3] Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. It's all connected: A journey through test-time memorization, attentional bias, retention, and online optimization, 2025.

[4] Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time, 2024.

[5] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer, 2020.

[6] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers, 2022.

[7] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.

[8] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022.

[9] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality, 2024.

[10] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024.

[11] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces, 2022.

[12] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer, 2020.

[13] James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. Fnet: Mixing tokens with fourier transforms, 2022.

[14] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, Xuzheng He, Haowen Hou, Jiaju

Lin, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartlomiej Koptyra, Hayden Lau, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Bolun Wang, Johan S. Wind, Stanislaw Wozniak, Ruichong Zhang, Zhenyuan Zhang, Qihang Zhao, Peng Zhou, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. Rwkv: Reinventing rnns for the transformer era, 2023.

[15] Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y. Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models, 2023.

[16] Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling, 2019.

[17] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.

[18] Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo,

Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): Rnns with expressive hidden states, 2025.

[19] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models, 2023.

[20] Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. Synthesizer: Rethinking self-attention in transformer models, 2021.

[21] Max Vladymyrov, Johannes von Oswald, Mark Sandler, and Rong Ge. Linear transformers are versatile in-context learners, 2024.

[22] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity, 2020.

[23] Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training, 2024.

[24] Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences, 2021.