The End of Transformers? On Challenging Attention and the Rise of Sub-Quadratic Architectures

Anonymous ACL submission

Abstract

Transformers have dominated sequence processing tasks for the past seven years—most notably language modeling. However, the inherent quadratic complexity of their attention mechanism remains a significant bottleneck as context length increases. This paper surveys recent efforts to overcome this bottleneck, including advances in (sub-quadratic) attention variants, recurrent neural networks, state space models, and hybrid architectures. We critically analyze these approaches in terms of compute and memory complexity, benchmark results, and fundamental limitations to assess whether the dominance of pure-attention transformers may soon be challenged.

1 Introduction

The transformer architecture represents a foundational breakthrough in *Natural Language Processing* (NLP) (Vaswani et al., 2017), forming the backbone of most *Large Language Models* (LLMs) (Brown et al., 2020) and serving as a reliable architecture choice for predictable performance scaling laws (Kaplan et al., 2020; Hoffmann et al., 2022). Its self-attention mechanism (Bahdanau et al., 2015) projects inputs into *queries* (Q), *keys* (K), and *values* (V), enabling efficient pairwise token interactions:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

Despite providing direct $\mathcal{O}(1)$ paths between any pair of tokens, computing the full $n \times n$ attention matrix incurs $\mathcal{O}(n^2)$ time complexity, increasing latency and compute costs as the input length n grows (Vaswani et al., 2017). This has motivated research efforts into sub-quadratic sequence-modeling operators to replace attention, aiming to improve efficiency while retaining strong task performance. These include sub-quadratic attention variants (Katharopoulos et al., 2020), *Recurrent Neural Networks* (RNNs) (Beck et al., 2024), *State Space Models* (SSMs) (Gu and Dao, 2023; Gu et al., 2022), and hybrids thereof (De et al., 2024).

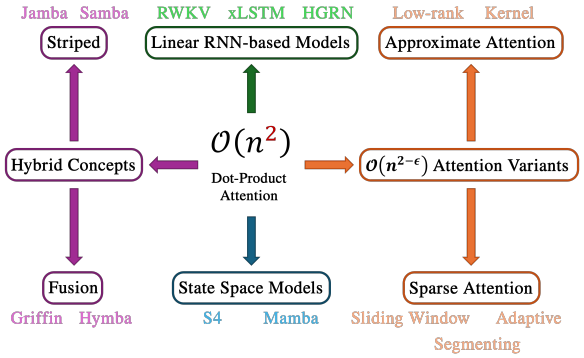


Figure 1: The four types of dot-product attention alternatives identified in our survey, including examples for each type. We further distinguish between two major classes for hybrid concepts, namely striped and fusion hybrids, as well as for sub-quadratic attention variants, namely approximate and sparse attention.

This paper reviews alternatives to transformers and examines whether their dominance may soon be challenged. Our main contributions are:

(1) A systematic review of the most relevant (sub)-quadratic attention variants, RNNs, SSMs, and hybrid architectures. An overview can be found in Figure 1.

(2) A comparative analysis of time and memory complexity for training and inference of sequence-modeling mechanisms, as well as reported benchmark results for SOTA models.

(3) A critical analysis of strengths, tradeoffs, and limitations, with an informed perspective on when and where pure attention-based transformers may be surpassed.

Our methodology is described in Appendix A.1.

2 Related Review Work

While several recent and concurrent works overlap with aspects of our scope, they differ in focus and conclusions. For example, Schneider (2025) discusses hypothetical post-transformer architectures without restricting to sub-quadratic complexity or state-of-the-art performance. Wang et al. (2024c) reviews approaches for handling longer input sequences, and Tiezzi et al. (2025) examines alternative architectures from the perspective of recurrent processing.

Several surveys provide overviews of techniques for efficient transformers and LLMs in general (Tay et al., 2022; Wan et al., 2024; Miao et al., 2024; Tang et al., 2024; Miao et al., 2023; Huang et al., 2023), but these often emphasize linear attention variants when considering alternative architectures. There are also focused surveys on specific subgroups, such as SSMs (Somvanshi et al., 2025; Wang et al., 2024b) and recurrent models (Tiezzi et al., 2024). Some works address models for domains like computer vision (Patro and Agneeswaran, 2024) or time series forecasting (Kim et al., 2025), whereas our emphasis is on NLP tasks and sub-quadratic alternatives to attention-based models.

Finally, Strobl et al. (2024) provide a detailed overview of previous works on transformer expressivity, which relates to our discussion of architectural limitations in Section 8.

3 $\mathcal{O}(n^2)$ Attention Variants

Despite not breaking the $\mathcal{O}(n^2)$ bottleneck, many attention variants deliver substantial practical speedups with no reduction in quality compared to standard attention.

Reducing KV Cache To reduce unnecessary recomputations, the keys and values of attention are often cached during inference. Managing such a *key-value* (KV) cache efficiently is key for reducing memory requirements. *Multi-Query Attention* (MQA) (Shazeer, 2019) and *Grouped-Query Attention* (GQA) (Ainslie et al., 2023) share key and value matrices across attention heads, reducing cache size by a constant factor but at the cost of reduced expressivity. *Multi-Head Latent Attention* (MLA), introduced by DeepSeek (DeepSeek-AI et al., 2024; DeepSeek-AI et al., 2025), uses a shared latent matrix among heads, which is projected back individually, achieving similar cache savings but with better performance than MQA and GQA. Refer to Li et al. (2025b) and Luohe et al. (2024) for a more detailed overview of KV cache techniques.

Flash Attention FlashAttention (Dao et al., 2022) and its successors exploit GPU memory hierarchies to make attention both faster and more memory-efficient, reducing memory usage to be linear in sequence length and delivering 2–4× runtime speedups over strong baselines. FlashAttention-2 (Dao, 2023) improved thread work partitioning for further speedup (as proven by GPT-style (Brown et al., 2020) LLM training), while FlashAttention-3 (Shah et al., 2024), specialized for Hopper GPUs, adds asynchrony and low-precision operations for an additional 1.5–2× boost.

Paged Attention Paged Attention (Kwon et al., 2023) improves inference memory efficiency by partitioning the KV cache into fixed-size pages and tracking them via a page table, boosting throughput 2–4× and eliminating padding.

4 Sub-Quadratic Architectures

Categorizing sub-quadratic attention alternatives is challenging due to overlapping ideas and mechanisms. We organize them as Linear Attention, Recurrent Models, SSMs, and Hybrids according to their main design motivation, though some (e.g., RWKV-7) fall into several categories. Earlier sub-quadratic architectures now outperformed are listed in Appendix A.2 for completeness.

4.1 $\mathcal{O}(n^{2-\epsilon})$ Attention Variants

Approximate Attention Approximate attention mechanisms, including linear attention, reduce computational cost by using approximations such as kernel functions or low-rank factorization. Kernel-based linear attention reformulates self-attention as a linear dot-product in feature space, achieving $\mathcal{O}(n)$ complexity (Katharopoulos et al., 2020; Zhuoran et al., 2021), but may suffer from reduced expressivity if the kernel is poorly chosen. Sequential cumulative summation can also slow inference in causal settings (e.g., Linear Transformer (Katharopoulos et al., 2020), Performer (Choromanski et al., 2020)). Low-rank methods—e.g., Linformer (Wang et al., 2020)—similarly achieve $\mathcal{O}(n)$ complexity, but their effectiveness depends on the rank selected.

Recent variants such as REGAL (Lu et al., 2025), Hedgehog (Zhang et al., 2024), and RoFly (Ro et al., 2025) further improve efficiency and expressivity. Log-linear attention (Guo et al., 2025) extends linear attention by allowing a logarithmically growing set of hidden states, providing a flexible trade-off between efficiency and expressiveness.

Sparse Attention Sparse attention mechanisms focus computation on a subset of the sequence using fixed or learnable patterns. Sparse Transformers (Child et al., 2019) pioneered sparse factorizations of the attention matrix, reducing complexity to $\mathcal{O}(n\sqrt{n})$. Local (sliding window) attention restricts computation to a window around each token and is often paired with global attention, as in Longformer (Beltagy et al., 2020), to regain expressivity by allowing selected tokens to attend globally. Other variants, such as strided or random patterns, are often combined (e.g., Zaheer et al., 2020). While some sparse patterns can achieve $\mathcal{O}(n)$ time and memory complexity, they may underperform on tasks requiring fine-grained global dependencies and often require task-specific tuning. Learnable and adaptive sparsity patterns (e.g., Correia et al., 2019) are proposed to address these limitations.

Lightning Attention Lightning Attention—also known as Lightning Attention-2 (Qin et al., 2024b)—divides attention into intra-block (standard attention) and inter-block (linear attention via kernel tricks) computations. This "divide and conquer" strategy addresses the slow training of causal linear attention—caused by sequential cumulative summations—by combining efficient intra-block processing with fast, kernel-based inter-block calculations. Lightning Attention also incorporates IO-aware optimizations from FlashAttention and enhances GPU performance through tiling. Both forward and backward passes have time complexity $\mathcal{O}(nd^2)$ (Qin et al., 2024c). It is used by MiniMax-01 (Li et al., 2025a), who report that for a given computational budget, Lightning Attention models can use more parameters and tokens, achieving lower loss than models with standard softmax attention.

4.2 Linear RNN-based Models

Recurrent Neural Networks (RNNs) process sequences by maintaining a fixed-size state updated at each time step, allowing them to model temporal dependencies (Yu et al., 2019). *Long Short-Term Memory* (LSTM) networks (Hochreiter and Schmidhuber, 1997) mitigate the vanishing gradient problem through a complex gating mechanism, while *Gated Recurrent Units* (GRU) (Cho et al., 2014) offer a simpler alternative with similar performance and lower computational cost.

RNNs and their variants offer linear autoregressive generation, but suffer from (1) varying degrees of vanishing/exploding gradients, (2) limited training parallelism, and (3) lack of expressivity due to a representation state not scaling with context length (Yu et al., 2019).

Receptance Weighted Key Value (RWKV) RWKV-4 (Peng et al., 2023) builds on the *Attention Free Transformer* (AFT) (Zhai et al., 2021) by using channel-wise time decay vectors in place of global interaction weights, effectively transforming linear attention into an RNN. Training has a complexity of $\mathcal{O}(Bnd^2)$, involving an attention-like WKV computation of $\mathcal{O}(Bnd)$ (with trainable decay vector W, key K, and value V), parallelizable over batch (B) and hidden dimension (d), but not the sequence length n. A custom CUDA kernel was developed to further improve the efficiency of the computations. Inference resembles an RNN but includes channel- and sequence-mixing, utilizing both previous input and hidden state. With these architectural tweaks, RWKV combines transformer-like scaling laws, competitive performance, and lower inference costs, but inherits limitations of recurrence, such as sensitivity to input order and reduced recall (see Section 8.2). The latest version, Goose (RWKV-7) (Peng et al., 2025), introduces a generalized delta rule, vector-valued gating, in-context learning rates, and a relaxed value replacement rule. RWKV-7 offers constant memory and inference time per token, parallelizable training, and increased expressivity beyond TC^0 transformers (see Section 8.1). See Li et al. (2025c) for a detailed overview.

Although RWKV-7 incorporates attention-inspired mechanisms and could be viewed as a hybrid, we classify it as the current SOTA in RNN-based models.

Hierarchically Gated Recurrent Neural Network (HGRN) HGRN (Qin et al., 2023) consists of stacked layers comprising token mixing (HGRU) and channel mixing (GLU) modules. Unlike S4 or RWKV-4, HGRN uses data-dependent, dynamic decay rates via forget gates, allowing lower layers

to focus on short-term and higher layers on long-term dependencies. Learnable lower bounds on forget gates prevent vanishing gradients.

To address limited recurrent state size, HGRN2 (Qin et al., 2024d) expands the state non-parametrically, improving scaling and outperforming Mamba on Long Range Arena (Tay et al., 2021), though pretrained transformers like LLaMA (Touvron et al., 2023) still perform better on long-context tasks. HGRN2 has been scaled to 3B parameters.

xLSTM xLSTM (Beck et al., 2024) enhances the LSTM architecture by incorporating state expansion, exponential gating, normalization, and stabilization techniques. It stacks two specialized LSTM modules: sLSTM, with scalar memory and update mechanisms for efficient state mixing and tracking, and mLSTM, with matrix memory and a covariance-based update rule for improved memorization and parallelism. The mLSTM's matrix memory supports tasks like Multi-Query Associative Recall. xLSTM achieves linear time and constant memory complexity, but incurs additional overhead from complex memory operations, partially offset by hardware-aware optimizations.

4.3 State Space Models

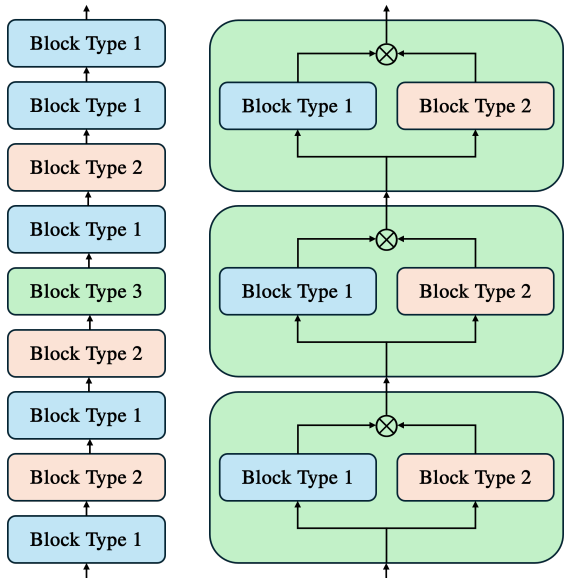
State Space Models (SSMs), originally from control theory for modeling dynamic systems via state variables, have emerged as promising sub-quadratic alternatives to transformers. A key aspect is their dual perspective: a recurrent formulation enables $\mathcal{O}(n)$ inference, while a convolutional view allows for $\mathcal{O}(n \log(n))$ training via efficient FFT-based convolutions.

Structured SSMs Structured SSMs impose a specific mathematical structure—such as low-rank or diagonal-plus-low-rank forms—on state transition and input matrices, enabling efficient and expressive modeling of long-range dependencies. S4 (Gu et al., 2022) introduces the use of a *Highly Predictive Polynomial Projection Operator* (HiPPO) matrix for initializing the state transition. This approach enables the construction of global convolution kernels that can efficiently encode long-term dependencies. At the time of release, S4 matched the performance of transformers (Gu et al., 2022). S5 (Smith et al., 2023) simplifies and extends S4 by replacing its diagonal block structure with dense matrices. Additionally, S5 leverages an efficient parallel scan, removing the need for S4's convolutional and frequency domain computations and streamlining kernel computation.

Selective SSMs Mamba (Gu and Dao, 2023) advances SSMs by replacing fixed transition matrices with input-dependent functions, increasing flexibility and expressivity. Its core is the Mamba block, which combines the ideas of H3 (Fu et al., 2022) and gated MLP blocks by adding a convolution and an SSM to the main branch of the gated MLP. Efficient implementation is achieved via kernel fusion, parallel scan, and recomputation.

Mamba2 (Dao and Gu, 2024) further unifies structured SSMs with attention mechanisms, enabling the application of transformer-style optimizations. It uses modified Mamba blocks for tensor parallelism and introduces the *State Space Dual* (SSD) layer as the inner SSM, which, in its recurrent form, is a selective SSM with single-input single-output structure. This design slightly reduces expressivity but significantly improves training efficiency on modern accelerators.

5 Hybrids



(a) Striped Hybrid (b) Fusion Hybrid

Figure 2: Different types of hybrids. (a): block types using different primitives are connected in series. (b): block types are connected in parallel.

Hybrid architectures combine different primitives—such as SSMs, attention, and RNNs—to leverage their strengths while mitigating the limitations of individual approaches (see

Sections 8.1 and 8.2). Such hybrids are usually of a striped (i.e., alternating primitives in series) or a fusion nature (i.e., primitives are calculated in parallel, combining their outputs). See Figure 2 for reference.

5.1 $\mathcal{O}(n^2)$ Hybrids

SSM + Attention Recent studies show that combining SSM and attention layers often outperforms using either one alone. For instance, Dao and Gu (2024) demonstrated that integrating SSD layers, attention, and MLPs can surpass pure Transformers and Mamba-2. Jamba (Lenz et al., 2025) merges transformer, Mamba, and *Mixture-of-Experts* (MoE) layers into a striped hybrid, achieving performance comparable to Llama-2 70B and Mixtral, but with 2x–7x longer context windows, 3x higher throughput, fewer total parameters (52B, 12B active), and reduced KV cache memory (32GB for 256K tokens vs. 4GB for Mixtral). Another notable example is the MambaFormer (Park et al., 2024), another striped hybrid.

Lightning Attention + Attention Li et al. (2025a) introduces the MiniMax-01 series by combining lightning attention with an MoE approach. To address lightning attention's limited retrieval, Hybrid-lightning replaces lightning attention with $\mathcal{O}(n^2)$ attention every eight layers, resulting in a striped hybrid. MiniMax-Text-01 was competitive with SOTA models like GPT-4o and Claude-3.5-Sonnet at the time of release, supporting context windows up to 1M tokens during training and 4M during inference at reasonable cost. However, it still struggles with multilevel instruction following due to sparse training data.

5.2 $\mathcal{O}(n^{2-\epsilon})$ Hybrids

De et al. (2024) propose the *Real-Gated Linear Recurrent Unit* (RG-LRU), a gated LRU (Orvieto et al., 2023) variant without complex transformations in the recurrence as they do not improve language modeling in practice. RG-LRU, a fusion hybrid of local attention and linear recurrence, is used for sequence mixing in a recurrent block, replacing MQA.

Griffin, using RG-LRU, achieves higher inference throughput and lower latency on long sequences than MQA Transformers (De et al., 2024). On benchmarks, Griffin-3B outperforms Mamba-3B, and Griffin-7B and 14B are competitive with Llama-2 despite using much less training data. Griffin is also used as the base for RecurrentGemma (Botev et al., 2024).

Other notable sub-quadratic hybrids include Hymba (Dong et al., 2025), combining both fusion and striped hybrid patterns, and Samba (Ren et al., 2025), a striped hybrid, both using a combination of sliding window attention and Mamba/SSM layers.

6 Novel Architecture Design Concepts

Memory System Design Recent models increasingly integrate several memory types (Irie et al., 2025; Nunez et al., 2025). Titans (Behrouz et al., 2024) introduce meta in-context neural long-term memory, storing surprising data at test time, and combine core attention-based short-term, neural long-term, and persistent task memory modules.

B'MOJO (Zancato et al., 2025) generalizes transformers and SSMs by blending permanent, short-term, fading, and long-term memories, with a sliding attention mechanism to aggregate information. Both models show good results versus transformers on several benchmarks (see Table 2).

Tailored Architecture Search Thomas et al. (2024)'s STAR framework unifies popular sequence model architectures under the theory of *Linear Input-Varying systems* (LIVs), creating a larger and more structured search space for model design. Given target metrics such as cache size, perplexity, or device latency, STAR uses gradient-free evolutionary algorithms to automatically search the LIV space and generate architectures optimized for several objectives, outperforming highly-tuned transformer and hybrid models on various quality and efficiency frontiers. One of the first models realized through STAR (although with slight modifications) is the strong edge model LFM2 (LiquidAI, 2025).

7 Complexity and Benchmark Analysis

Moving away from the qualitative analysis in the previous sections, this section focuses on quantitative results and a direct comparison of model architectures in terms of complexity and benchmark performance.

Complexity Comparison We compare the complexities of selected sequence-modeling mechanisms in Table 1. It is important to note that these complexities are sometimes dominated by feed-forward neural networks in the full model, e.g., in

Method	Training			Inference	
	Time	Space	Parallel	Time	Space
FFT-Convolution	$\mathcal{O}(Bnd\log(dn))$	$\mathcal{O}(Bnd)$	Yes	$\mathcal{O}(nd\log(nd))$	$\mathcal{O}(nd)$
RNN	$\mathcal{O}(Bnd^2)$	$\mathcal{O}(Bnd)$	No	$\mathcal{O}(d^2)^2$	$\mathcal{O}(nd)$
Vanilla Transformer	$\mathcal{O}(B(n^2d+nd^2))$	$\mathcal{O}(B(n^2+nd))$	Yes	$\mathcal{O}(n^2d+d^2n)$	$\mathcal{O}(n^2+nd)$
LSH (Reformer)	$\mathcal{O}(Bd^2n\log n)$	$\mathcal{O}(Bn\log n+Bnd)$	Yes	$\mathcal{O}(d^2n\log n)$	$\mathcal{O}(n\log n+nd)$
FAVOR+ (Performer)	$\mathcal{O}(Bnd^2\log d)$	$\mathcal{O}(Bnd\log d+Bd^2\log d)$	Yes	$\mathcal{O}(nd^2\log d)$	$\mathcal{O}(nd\log d+d^2\log d)$
Linear Transformer	$\mathcal{O}(Bnd^2)$	$\mathcal{O}(B(nd+d^2))$	Yes	$\mathcal{O}(nd^2)$	$\mathcal{O}(nd+d^2)$
Lightning Attention	$\mathcal{O}(Bnd^2)$	$\mathcal{O}(B(nd+d^2))$	Yes	$\mathcal{O}(nd^2)$	$\mathcal{O}(nd+d^2)$
RWKV	$\mathcal{O}(Bnd^2)$	$\mathcal{O}(Bnd)$	Yes	$\mathcal{O}(nd)$	$\mathcal{O}(d)$
Hyena-3	$\mathcal{O}(Bnd\log(dn))$	$\mathcal{O}(Bnd)$	Yes	$\mathcal{O}(nd\log(n+d))$	$\mathcal{O}(nd)$
S4	$\mathcal{O}(Bnd\log(dn))$	$\mathcal{O}(Bnd)$	Yes	$\mathcal{O}(d^2)$	$\mathcal{O}(nd)$
Mamba[3]	$\mathcal{O}(B(nd^2+nd\log(nd)))$	$\mathcal{O}(Bnd)$	Yes	$\mathcal{O}(nd^2+nd\log(nd))$	$\mathcal{O}(nd)$

Table 1: Overview on time & space complexities for training on a single batch and inference of a single token of different sequence-modeling mechanisms. n: sequence length; d: hidden dimension; B: batch size

S4, which have a time complexity of $\mathcal{O}(nd^2)$. Except for RWKV, which can process a single query at a time at inference, models are lower bounded on memory complexity by storing the sequence in its entirety. Many of these algorithms rely on projections, thus requiring at least $\mathcal{O}(nd^2)$ operations, often serving as an upper bound for time complexity. Another major influence on time complexity is the use of FFT convolutions, as used in SSM-based models for training, which requires $\mathcal{O}(nd\log(dn))$ computational steps, binding the algorithm to log-linear time.

7.1 Benchmark Performance

In Table 2, we provide a performance comparison of previously mentioned sub-quadratic models with recent high-performing models based on quadratic attention. We chose a configuration variety that sees frequent use: two table sections comparing models with a total parameter size of 0.7-1.5B and 14-70B (for MoE models, the total parameter count applies) on eight prominent benchmarks that cover a broad range of downstream tasks. For the model and benchmark sources, see Appendix A.4.

We can see that in a low-parameter setting (0.7-1.5B), several edge models compete for the top scores. In particular, Samba and RWKV7-World3 significantly outperform the full attention Llama 3.2 and Qwen2.5 in several instances. In the midrange (14-70B), no pure sub-quadratic models are present anymore; merely the hybrids Griffin and Jamba remain, with only the latter realistically competing with Qwen2.5 and Llama3.1. In the evaluation of frontier (100B+) models, we referred to the LMsys chatbot arena (Chiang et al., 2024) instead of a custom-made table. Across all benchmarks[1], only MiniMax-Text-01 (Li et al., 2025a) appears in the top-20 ranking once, but among the top 10, we cannot find any single model known to be built on an alternative architecture.

8 Fundamental Architectural Limitations

Both quadratic attention and sub-quadratic architectures face fundamental limitations that cannot be overcome by scaling parameters or training. In this section, we discuss these inherent restrictions. Broader limitations of language models in general (e.g., Wheeler and Jeunen, 2025) are beyond this survey's scope.

8.1 Limitations of Attention

General Theoretical Expressivity The standard transformer forward pass belongs to the log-time uniform TC^0 circuit complexity class (Merrill and Sabharwal, 2023). This fundamentally limits its ability to simulate finite automata or solve graph connectivity—necessary for state tracking and multi-step reasoning (Merrill and Sabharwal, 2025). In practice, such tasks are tractable for short contexts (e.g., by using transformers of depth $\mathcal{O}(\log C)$ for context length C), but remain infeasible for unbounded inputs under standard complexity assumptions. To scale up these capabilities, the model dimension must grow with the task complexity, as is also highlighted in related work (Hahn, 2020; Sanford et al., 2023).

Allowing intermediate steps, i.e., *Chain of Thought* (CoT) (Wei et al., 2022), increases transformer expressivity w.r.t. the number of steps. Li

[1] Accessed on 2025-07-25

[2] Assuming the sequence has been processed already, only necessary once

[3] We consider an entire Mamba layer here, including projections

Model	Benchmark Selection							
0.7-1.5B	Size	**MMLU**	**LMB**	**ARC-E**	**ARC-C**	**Wino.**	**Hella.**	**PIQA**
Titans-MAG	760M	-	41.0	68.2	36.2	52.9	48.9	70.3
Griffin	1B	29.5	-	67.0	36.9	65.2	67.2	**77.4**
Llama3.2*	1B	32.1	63.0	-	-	60.7	63.7	-
HGRN2	1.3B	-	49.4	58.1	28.1	52.3	51.8	71.4
Mamba2	1.3B	-	65.7	61.0	33.3	60.9	59.9	73.2
xLSTM[1:0]	1.3B	-	57.8	64.3	32.6	60.6	60.9	74.6
BMoJo-Fading	1.4B	-	45.4	52.3	26.6	53.3	46.0	70.0
RWKV7-World3	1.5B	43.3	**69.5**	78.1	44.5	68.2	**70.8**	77.1
Qwen2.5*	1.5B	**60.9**	63.0	75.5	**54.7**	65.0	67.9	75.8
Samba	1.7B	48.0	-	**79.3**	48.2	**72.9**	49.7	77.1
14-70B	Size	**MMLU**	**BBH**	**GSM8K**	**ARC-C**	**Wino.**	**Hella.**	**HumanEval**
Griffin	14B	49.5	-	-	50.8	74.1	81.4	-
Qwen*	14B	79.7	78.2	90.2	67.3	81.0	84.3	56.7
Jamba	52B	67.40	45.40	59.9	64.40	82.5	87.1	29.30
Mixtral*	56B	70.6	-	60.4	59.7	77.2	84.4	40.2
Llama3.1*	70B	79.5	81.0	95.1	68.8	**85.3**	**88.0**	48.2
Qwen2.5*	72B	**86.1**	**86.3**	**95.8**	**72.4**	83.9	87.6	**59.1**

Table 2: Performance comparison of recent pure quadratic attention LMs (highlighted with *) and subquadratic models of similar size. Best results for each parameter category are marked in **bold**, second-best results are underlined. Model names are in bold or underlined when they scored first or second at least once. Results are rounded to one decimal point. For sources, see Appendix A.4

et al. (2024) show that with T CoT steps, constant-depth transformers with $\mathcal{O}(\log n)$ embeddings can solve any problem solvable by boolean circuits of size T. Additionally, Qiu et al. (2025) prove that prompting is Turing-complete: for any computable function, a finite-size transformer can compute it with an appropriate prompt. However, these enhancements also introduce new drawbacks, as shown by Amiri et al. (2025); Peng et al. (2024); Saparov et al. (2025).

Length Generalization Transformers struggle to extrapolate, i.e., to generalize from shorter training context sizes to longer test sequences. In addition to being limited by memory constraints, the transformer architecture has fundamental length-generalization limits caused by positional encodings (Kazemnejad et al., 2023). While transformers without position encodings (NoPE) seem to be an alternative and work for longer sequences than explicit encodings, they still impose a context length limit (Wang et al., 2024a).

Building upon Huang et al. (2025)'s framework to analyze length generalization, Veitsman et al. (2025) show that, if pretraining is done right, certain capabilities w.r.t. length generalization of transformers can be improved, but fundamental limitations persist. For models like SSMs and B'MOJO, the length generalization is instead limited by the capacity of the recurrent state.

For the framework of Huang et al. (2025) and a more detailed analysis of the limitations of attention, see Appendix A.3.

8.2 Limitations of Sub-Quadratic Alternatives

Sub-quadratic architectures share some limitations with quadratic attention. For instance, Merrill et al. (2024) showed that SSMs are also limited to the complexity class TC^0. Although these models improve efficiency, they introduce new challenges due to the inherent difficulty of compressing sequence context into a reduced state.

This finite state capacity has strong implications for "lookup table" tasks (e.g., MQAR (Arora et al., 2024a), hop_k (Sanford et al., 2024)), where such information is part of the input, as SSMs cannot recall an arbitrary amount of information previously seen Arora et al. (2024b); De et al. (2024); Jelassi et al. (2024), even though recent work (Grazzi et al., 2024) shows that some improvements can be made, as seen in Mamba (Gu and Dao, 2023).

A similar problem occurs in linear RNNs, which are highly sensitive to the order of context, making prompt engineering critical—selection and recall become much harder as input order varies (Sutskever et al., 2014; Arora et al., 2024c). RNNs require $\Omega(N)$ space for reliable recall (Arora et al., 2024b), and constant-memory models cannot perform associative recall or solve tasks like q-sparse averaging or copying, unlike shallow transformers

(Sanford et al., 2024; Jelassi et al., 2024; Wen et al., 2025).

Han et al. (2025) show that linear attention is not injective, often assigning identical attention weights to different queries and causing semantic confusion. They also demonstrate that linear attention struggles with effective local modeling, a strength of softmax attention. Related work finds that the low-rank nature of linear attention's feature map can further hinder modeling of complex spatial or local information (Fan et al., 2025).

Backurs and Indyk (2018) prove that under the SETH (which implies $P \neq NP$), edit distance cannot be computed in truly subquadratic time, setting a fundamental limit on sequence comparison efficiency for any such architecture. Under the same assumption, Alman and Yu (2025) show that document similarity tasks inherently require quadratic time.

Implications The limitations applying to alternative architectures mostly subsume the limitations applying to transformers. This implies that while sub-quadratic alternatives significantly enhance efficiency and lower computational costs, they do not fundamentally surpass transformers in theoretical expressivity.

9 Discussion

In this section, we synthesize insights from our review to discuss whether sub-quadratic and hybrid alternatives start claiming meaningful territory.

9.1 Current Landscape

Despite the reviewed advances in alternative architectures, at the time of writing, most frontier general-purpose models strongly rely on full attention mechanisms. No model scoring in the top 10 on LLMSys (Chiang et al., 2024) is known to be sub-quadratic or a hybrid, showing that the "Transformer++" remains the default choice when compute is not a limiting factor. We have also seen that full attention is free from many limitations that apply to alternative architectures (Section 8.2), adding to the extent of their superiority.

However, the picture changes for edge models, where compute, memory, and latency are tightly bound, and alternative architectures have gained substantial traction. Especially hybrids, such as Samba (Ren et al., 2025) or RWKV7 (Peng et al., 2025), offer favorable inference properties. They can meet resource constraints by offloading local or intermediate computations to more efficient modules, while maintaining reasonable generalization and global context modeling via attention. For the edge, we also increasingly see differentiated memory modeling with newer models, like Titans (Behrouz et al., 2024) and B'MOJO (Zancato et al., 2025), segmenting memory into short-term, long-term, and permanent storage, assigning specialized mechanisms to each.

In the mid-size regime, hybrids like Jamba (Lenz et al., 2025) show promise, though they remain a minority and do not outperform well-tuned transformers. Their advantages are domain-specific, tied to scenarios where efficiency provides tangible gains. In general, the maturity of transformer infrastructure also makes switching to other architectures costly due to ecosystem inertia (Rahman et al., 2025; Brem and Nylund, 2024). However, work that enables the conversion of pretrained transformers to alternative architectures without retraining, such as RWKV, starts lowering these barriers.

Together, these trends signal a shift toward architectural diversity. While transformers remain dominant, alternatives are finding footholds in specific use cases and operational niches.

9.2 Outlook

At the frontier, full attention is likely to remain central for the foreseeable future. Still, even these models may begin incorporating hybrid elements, especially for memory management or task-specific routing. In this sense, we also anticipate model routing and *Mixture of Architectures* (MoA) paradigms to become more relevant. The shift is not toward replacement, but toward building flexible systems from a growing set of specialized primitives. This idea has already been surfaced by Yu et al. (2025) and Varangot-Reille et al. (2025), although they focused more on model sizes than underlying architectures.

10 Conclusion

Through our review of recent subquadratic architectures, we have highlighted the most promising alternatives to full attention for sequence modeling in NLP. Our analysis shows that these models introduce valuable tradeoffs in efficiency and latency, particularly in edge and mid-sized deployments. However, they remain fundamentally constrained in generality compared to transformers and will not compete in the frontier for the foreseeable future.

Limitations

As a focused and concise survey, our work comes with several limitations. We restrict our analysis to language models, and therefore, our findings may not generalize to other modalities such as vision, audio, or multimodal systems. Additionally, the performance comparison presented in Table 2 is limited in its language coverage, as it focuses primarily on English. There is also a slight variation in training data and procedure across the benchmark results of the models we report on, which is explained in A.4. Finally, while our methodology (see Appendix A.1) reflects a rigorous effort to identify and synthesize relevant literature, researchers with a different focus could consider some missing works more significant.

Acknowledgments

AUTHOR INFORMATION REDACTED FOR BLIND SUBMISSION.

This work used LLM-based tools for language edits and clarity improvements. All analysis, research, and ideas are either our own or cited.

References

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. 2023. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901, Singapore. Association for Computational Linguistics.

Josh Alman and Hantao Yu. 2025. Fundamental limitations on subquadratic alternatives to transformers. In *The Thirteenth International Conference on Learning Representations*.

Alireza Amiri, Xinting Huang, Mark Rofin, and Michael Hahn. 2025. Lower bounds for chain-of-thought reasoning in hard-attention transformers. *CoRR*, abs/2502.02393.

Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Re. 2024a. Zoology: Measuring and improving recall in efficient language models. In *The Twelfth International Conference on Learning Representations*.

Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, James Zou, Atri Rudra, and Christopher Ré. 2024b. Simple linear attention language models balance the recall-throughput tradeoff. In *Proceedings of the 41st International Conference on Machine Learning*.

Simran Arora, Aman Timalsina, Aaryan Singhal, Sabri Eyuboglu, Xinyi Zhao, Ashish Rao, Atri Rudra, and Christopher Re. 2024c. Just read twice: closing the recall gap for recurrent language models. In *Workshop on Efficient Systems for Foundation Models II @ ICML2024*.

Arturs Backurs and Piotr Indyk. 2018. Edit distance cannot be computed in strongly subquadratic time (unless seth is false). *SIAM Journal on Computing*, 47(3):1087–1097.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. 2024. xlstm: Extended long short-term memory. In *Advances in Neural Information Processing Systems*, volume 37, pages 107547–107603. Curran Associates, Inc.

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. 2024. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*.

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

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI conference on artificial intelligence*, 34(05):7432–7439.

Aleksandar Botev, Soham De, Samuel L. Smith, Anushan Fernando, George-Cristian Muraru, Ruba Haroun, Leonard Berrada, Razvan Pascanu, Pier Giuseppe Sessa, Robert Dadashi, Léonard Hussenot, Johan Ferret, Sertan Girgin, Olivier Bachem, Alek Andreev, Kathleen Kenealy, Thomas Mesnard, Cassidy Hardin, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Armand Joulin, Noah Fiedel, Evan Senter, Yutian Chen, Srivatsan Srinivasan, Guillaume Desjardins, David Budden, Arnaud Doucet, Sharad Vikram, Adam Paszke, Trevor Gale, Sebastian Borgeaud, Charlie Chen, Andy Brock, Antonia Paterson, Jenny Brennan, Meg Risdal, Raj Gundluru, Nesh Devanathan, Paul Mooney, Nilay Chauhan, Phil Culliton, Luiz Gustavo Martins, Elisa Bandy, David Huntsperger, Glenn Cameron, Arthur Zucker, Tris Warkentin, Ludovic Peran, Minh Giang, Zoubin Ghahramani, Clément Farabet, Koray Kavukcuoglu, Demis Hassabis, Raia Hadsell, Yee Whye Teh, and Nando de Frietas. 2024. RecurrentGemma: Moving past transformers for efficient open language models.

Alexander Brem and Petra Nylund. 2024. The inertia of dominant designs in technological innovation: An

ecosystem view of standardization. *IEEE Transactions on Engineering Management*, 71:2640–2648.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Kyunghyun Cho, B van Merrienboer, Caglar Gulcehre, F Bougares, H Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*.

K. Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy J. Colwell, and Adrian Weller. 2020. Rethinking attention with performers. *ArXiv*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Gonçalo M. Correia, Vlad Niculae, and André F. T. Martins. 2019. Adaptively sparse transformers. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2174–2184.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359.

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

Soham De, Samuel L. Smith, Anushan Fernando, Aleksandar Botev, George Cristian-Muraru, Albert Gu, Ruba Haroun, Leonard Berrada, Yutian Chen, Srivatsan Srinivasan, Guillaume Desjardins, Arnaud Doucet, David Budden, Yee Whye Teh, Razvan Pascanu, Nando De Freitas, and Caglar Gulcehre. 2024. Griffin: Mixing gated linear recurrences with local attention for efficient language models.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan

Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning.

DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao Yang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jin Chen, Jingyang Yuan, Junjie Qiu, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruizhe Pan, Runxin Xu, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Size Zheng, T. Wang, Tian Pei, Tian Yuan, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Liu, Xin Xie, Xingkai Yu, Xinnan Song, Xinyi Zhou, Xinyu Yang, Xuan Lu, Xuecheng Su, Y. Wu, Y. K. Li, Y. X. Wei, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Zheng, Yichao Zhang, Yiliang Xiong, Yilong Zhao, Ying He, Ying Tang, Yishi Piao, Yixin Dong, Yixuan Tan, Yiyuan Liu, Yongji Wang, Yongqiang Guo, Yuchen Zhu, Yuduan Wang, Yuheng Zou, Yukun Zha, Yunxian Ma, Yuting Yan, Yuxiang You, Yuxuan Liu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhewen Hao, Zhihong Shao, Zhiniu Wen, Zhipeng Xu, Zhongyu Zhang, Zhuoshu Li, Zihan Wang, Zihui Gu, Zilin Li, and Ziwei Xie. 2024. DeepSeek-v2: A strong, economical, and efficient mixture-of-experts language model.

Xin Dong, Yonggan Fu, Shizhe Diao, Wonmin Byeon, ZIJIA CHEN, Ameya Sunil Mahabaleshwarkar, Shih-Yang Liu, Matthijs Van keirsbilck, Min-Hung Chen, Yoshi Suhara, Yingyan Celine Lin, Jan Kautz, and Pavlo Molchanov. 2025. Hymba: A hybrid-head architecture for small language models. In *The Thirteenth International Conference on Learning Representations*.

Qihang Fan, Huaibo Huang, and Ran He. 2025. Breaking the low-rank dilemma of linear attention. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25271–25280.

Daniel Y Fu, Tri Dao, Khaled K Saab, Armin W Thomas, Atri Rudra, and Christopher Ré. 2022. Hungry hungry hippos: Towards language modeling with state space models. *arXiv preprint arXiv:2212.14052*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Riccardo Grazzi, Julien Niklas Siems, Simon Schrodi, Thomas Brox, and Frank Hutter. 2024. Is mamba capable of in-context learning? In *Proceedings of the Third International Conference on Automated Machine Learning*, volume 256 of *Proceedings of Machine Learning Research*, pages 1/1–26. PMLR.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *ArXiv*, abs/2312.00752.

Albert Gu, Karan Goel, and Christopher Re. 2022. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*.

Han Guo, Songlin Yang, Tarushii Goel, Eric P Xing, Tri Dao, and Yoon Kim. 2025. Log-linear attention. *arXiv preprint arXiv:2506.04761*.

Michael Hahn. 2020. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171.

Dongchen Han, Yifan Pu, Zhuofan Xia, Yizeng Han, Xuran Pan, Xiu Li, Jiwen Lu, Shiji Song, and Gao Huang. 2025. Bridging the divide: reconsidering softmax and linear attention. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *CoRR*, abs/2009.03300.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

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, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Xinting Huang, Andy Yang, Satwik Bhattamishra, Yash Sarrof, Andreas Krebs, Hattie Zhou, Preetum Nakkiran, and Michael Hahn. 2025. A formal framework for understanding length generalization in transformers. In *The Thirteenth International Conference on Learning Representations*.

Yunpeng Huang, Jingwei Xu, Junyu Lai, Zixu Jiang, Taolue Chen, Zenan Li, Yuan Yao, Xiaoxing Ma, Lijuan Yang, Hao Chen, Shupeng Li, and Penghao Zhao. 2023. Advancing transformer architecture in long-context large language models: A comprehensive survey. *arXiv preprint*.

Kazuki Irie, Morris Yau, and Samuel J. Gershman. 2025. Blending complementary memory systems in hybrid quadratic-linear transformers.

Samy Jelassi, David Brandfonbrener, Sham M. Kakade, and Eran Malach. 2024. Repeat after me: transformers are better than state space models at copying. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

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

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR.

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan, Payel Das, and Siva Reddy. 2023. The impact of positional encoding on length generalization in transformers. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Jongseon Kim, Hyungjoon Kim, HyunGi Kim, Dongjun Lee, and Sungroh Yoon. 2025. A comprehensive survey of deep learning for time series forecasting: Architectural diversity and open challenges. *Artificial Intelligence Review*, 58:216.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Barak Lenz, Opher Lieber, Alan Arazi, Amir Bergman, Avshalom Manevich, Barak Peleg, Ben Aviram, Chen Almagor, Clara Fridman, Dan Padnos, Daniel Gissin, Daniel Jannai, Dor Muhlgay, Dor Zimberg, Edden M. Gerber, Elad Dolev, Eran Krakovsky, Erez Safahi, Erez Schwartz, Gal Cohen, Gal Shachaf, Haim Rozenblum, Hofit Bata, Ido Blass, Inbal Magar, Itay Dalmedigos, Jhonathan Osin, Julie Fadlon, Maria Rozman, Matan Danos, Michael Gokhman, Mor Zusman, Naama Gidron, Nir Ratner, Noam Gat, Noam Rozen, Oded Fried, Ohad Leshno, Omer Antverg, Omri Abend, Or Dagan, Orit Cohavi, Raz Alon, Ro'i Belson, Roi Cohen, Rom Gilad, Roman Glozman, Shahar Lev, Shai Shalev-Shwartz, Shaked Haim Meirom, Tal Delbari, Tal Ness, Tomer Asida, Tom Ben Gal, Tom Braude, Uriya Pumerantz, Josh Cohen, Yonatan Belinkov, Yuval Globerson, Yuval Peleg Levy, and Yoav Shoham. 2025. Jamba: Hybrid transformer-mamba language models. In *The Thirteenth International Conference on Learning Representations*.

Aonian Li, Bangwei Gong, Bo Yang, Boji Shan, Chang Liu, Cheng Zhu, Chunhao Zhang, Congchao Guo, Da Chen, Dong Li, et al. 2025a. Minimax-01: Scaling foundation models with lightning attention. *arXiv preprint arXiv:2501.08313*.

Haoyang Li, Yiming Li, Anxin Tian, Tianhao Tang, Zhanchao Xu, Xuejia Chen, Nicole Hu, Wei Dong, Qing Li, and Lei Chen. 2025b. A survey on large language model acceleration based on kv cache management.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. 2024. Chain of thought empowers transformers to

solve inherently serial problems. In *The Twelfth International Conference on Learning Representations*.

Zhiyuan Li, Tingyu Xia, Yi Chang, and Yuan Wu. 2025c. A survey of RWKV.

LiquidAI. 2025. Introducing LFM2: The fastest on-device foundation models on the market | liquid AI. Accessed: 2025-07-24.

Peng Lu, Ivan Kobyzev, Mehdi Rezagholizadeh, Boxing Chen, and Philippe Langlais. 2025. ReGLA: Refining gated linear attention. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2884–2898, Albuquerque, New Mexico. Association for Computational Linguistics.

Shi Luohe, Hongyi Zhang, Yao Yao, Zuchao Li, and hai zhao. 2024. Keep the cost down: A review on methods to optimize LLM's KV-cache consumption. In *First Conference on Language Modeling*.

William Merrill, Jackson Petty, and Ashish Sabharwal. 2024. The illusion of state in state-space models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 35492–35506. PMLR.

William Merrill and Ashish Sabharwal. 2023. A logic for expressing log-precision transformers. In *Advances in Neural Information Processing Systems*, volume 36, pages 52453–52463. Curran Associates, Inc.

William Merrill and Ashish Sabharwal. 2025. A little depth goes a long way: the expressive power of log-depth transformers.

Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, Xinhao Cheng, Hongyi Jin, Tianqi Chen, and Zhihao Jia. 2023. Towards efficient generative large language model serving: A survey from algorithms to systems.

Xupeng Miao, Shenhan Zhu, Fangcheng Fu, Ziyu Guo, Zhi Yang, Yaofeng Tu, Zhihao Jia, and Bin Cui. 2024. X-former elucidator: Reviving efficient attention for long context language modeling. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI 2024)*, volume 9, pages 8179–8187.

Elvis Nunez, Luca Zancato, Benjamin Bowman, Aditya Golatkar, Wei Xia, and Stefano Soatto. 2025. Expansion span: Combining fading memory and retrieval in hybrid state space models.

Antonio Orvieto, Samuel L Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. 2023. Resurrecting recurrent neural networks for long sequences. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*.

Jongho Park, Jaeseung Park, Zheyang Xiong, Nayoung Lee, Jaewoong Cho, Samet Oymak, Kangwook Lee, and Dimitris Papailiopoulos. 2024. Can mamba learn how to learn? a comparative study on in-context learning tasks. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Badri Narayana Patro and Vijay Srinivas Agneeswaran. 2024. Mamba-360: Survey of state space models as transformer alternative for long sequence modelling: Methods, applications, and challenges. *CoRR*, abs/2404.16112.

Binghui Peng, Srini Narayanan, and Christos Papadimitriou. 2024. On limitations of the transformer architecture. In *First Conference on Language Modeling*.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Gv, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan Wind, Stanisław Woźniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. 2023. RWKV: Reinventing RNNs for the transformer era. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077. Association for Computational Linguistics.

Bo Peng, Ruichong Zhang, Daniel Goldstein, Eric Alcaide, Xingjian Du, Haowen Hou, Jiaju Lin, Jiaxing Liu, Janna Lu, William Merrill, Guangyu Song, Kaifeng Tan, Saiteja Utpala, Nathan Wilce, Johan S. Wind, Tianyi Wu, Daniel Wuttke, and Christian Zhou-Zheng. 2025. RWKV-7 "goose" with expressive dynamic state evolution.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y. Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. 2023. Hyena hierarchy: towards larger convolutional language models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Zhen Qin, Dong Li, Weigao Sun, Weixuan Sun, Xuyang Shen, Xiaodong Han, Yunshen Wei, Baohong Lv, Xiao Luo, Yu Qiao, and Yiran Zhong. 2024a. Transnormerllm: A faster and better large language model with improved transnormer.

Zhen Qin, Weigao Sun, Dong Li, Xuyang Shen, Weixuan Sun, and Yiran Zhong. 2024b. Lightning attention-2: A free lunch for handling unlimited sequence lengths in large language models.

Zhen Qin, Weigao Sun, Dong Li, Xuyang Shen, Weixuan Sun, and Yiran Zhong. 2024c. Various lengths, constant speed: Efficient language modeling with lightning attention. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 41517–41535. PMLR.

Zhen Qin, Songlin Yang, Weixuan Sun, Xuyang Shen, Dong Li, Weigao Sun, and Yiran Zhong. 2024d. HGRN2: Gated linear RNNs with state expansion.

Zhen Qin, Songlin Yang, and Yiran Zhong. 2023. Hierarchically gated recurrent neural network for sequence modeling. In *Advances in Neural Information Processing Systems*, volume 36, pages 33202–33221. Curran Associates, Inc.

Ruizhong Qiu, Zhe Xu, Wenxuan Bao, and Hanghang Tong. 2025. Ask, and it shall be given: On the turing completeness of prompting. In *The Thirteenth International Conference on Learning Representations*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report.

Mohammad Shahedur Rahman, Peng Gao, and Yuede Ji. 2025. Hugginggraph: Understanding the supply chain of llm ecosystem. *arXiv preprint arXiv:2507.14240*.

Liliang Ren, Yang Liu, Yadong Lu, yelong shen, Chen Liang, and Weizhu Chen. 2025. Samba: Simple hybrid state space models for efficient unlimited context language modeling. In *The Thirteenth International Conference on Learning Representations*.

Yeonju Ro, Zhenyu Zhang, Souvik Kundu, Zhangyang Wang, and Aditya Akella. 2025. On-the-fly adaptive distillation of transformer to dual-state linear attention. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. 2023. Representational strengths and limitations of transformers. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. 2024. Transformers, parallel computation, and logarithmic depth. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Abulhair Saparov, Srushti Ajay Pawar, Shreyas Pimpalgaonkar, Nitish Joshi, Richard Yuanzhe Pang, Vishakh Padmakumar, Mehran Kazemi, Najoung Kim, and He He. 2025. Transformers struggle to learn to search. In *The Thirteenth International Conference on Learning Representations*.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. 2021. Linear transformers are secretly fast weight programmers. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9355–9366. PMLR.

Johannes Schneider. 2025. What comes after transformers? a selective survey connecting ideas in deep learninggpt. In *Agents and Artificial Intelligence: 16th International Conference, ICAART 2024, Rome, Italy, February 24–26, 2024, Revised Selected Papers, Part II*, page 55–82, Berlin, Heidelberg. Springer-Verlag.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. 2024. FlashAttention-3: Fast and accurate attention with asynchrony and low-precision.

Noam Shazeer. 2019. Fast transformer decoding: One write-head is all you need.

Jimmy T.H. Smith, Andrew Warrington, and Scott Linderman. 2023. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*.

Shriyank Somvanshi, Md Monzurul Islam, Mahmuda Sultana Mimi, Sazzad Bin Bashar Polock, Gaurab Chhetri, and Subasish Das. 2025. From s4 to mamba: A comprehensive survey on structured state space models.

Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. 2024. What formal languages can transformers express? a survey. *Transactions of the Association for Computational Linguistics*, 12:543–561.

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

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Yehui Tang, Yunhe Wang, Jianyuan Guo, Zhijun Tu, Kai Han, Hailin Hu, and Dacheng Tao. 2024. A survey on transformer compression.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2021. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2022. Efficient transformers: A survey. *ACM Comput. Surv.*, 55(6).

Armin W. Thomas, Rom Parnichkun, Alexander Amini, Stefano Massaroli, and Michael Poli. 2024. Star: Synthesis of tailored architectures.

Matteo Tiezzi, Michele Casoni, Alessandro Betti, Tommaso Guidi, Marco Gori, and Stefano Melacci. 2024. On the resurgence of recurrent models for long sequences – survey and research opportunities in the transformer era.

Matteo Tiezzi, Michele Casoni, Alessandro Betti, Tommaso Guidi, Marco Gori, and Stefano Melacci. 2025. Back to recurrent processing at the crossroad of transformers and state-space models. *Nature Machine Intelligence*, 7(5):678–688. Publisher: Nature Publishing Group.

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. 2023. Llama: Open and efficient foundation language models.

Clovis Varangot-Reille, Christophe Bouvard, Antoine Gourru, Mathieu Ciancone, Marion Schaeffer, and François Jacquenet. 2025. Doing more with less–implementing routing strategies in large language model-based systems: An extended survey. *arXiv preprint arXiv:2502.00409*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Yana Veitsman, Mayank Jobanputra, Yash Sarrof, Aleksandra Bakalova, Vera Demberg, Ellie Pavlick, and Michael Hahn. 2025. Born a transformer – always a transformer?

Zhongwei Wan, Xin Wang, Che Liu, Samiul Alam, Yu Zheng, Jiachen Liu, Zhongnan Qu, Shen Yan, Yi Zhu, Quanlu Zhang, Mosharaf Chowdhury, and Mi Zhang. 2024. Efficient large language models: A survey. *Transactions on Machine Learning Research*. Survey Certification.

Jie Wang, Tao Ji, Yuanbin Wu, Hang Yan, Tao Gui, Qi Zhang, Xuanjing Huang, and Xiaoling Wang. 2024a. Length generalization of causal transformers without position encoding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14024–14040, Bangkok, Thailand. Association for Computational Linguistics.

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

Xiao Wang, Shiao Wang, Yuhe Ding, Yuehang Li, Wentao Wu, Yao Rong, Weizhe Kong, Ju Huang, Shihao Li, Haoxiang Yang, Ziwen Wang, Bo Jiang, Chenglong Li, Yaowei Wang, Yonghong Tian, and Jin Tang. 2024b. State space model for new-generation network alternative to transformers: A survey.

Xindi Wang, Mahsa Salmani, Parsa Omidi, Xiangyu Ren, Mehdi Rezagholizadeh, and Armaghan Eshaghi. 2024c. Beyond the limits: a survey of techniques to extend the context length in large language models. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Kaiyue Wen, Xingyu Dang, and Kaifeng Lyu. 2025. RNNs are not transformers (yet): The key bottleneck on in-context retrieval. In *The Thirteenth International Conference on Learning Representations*.

Schaun Wheeler and Olivier Jeunen. 2025. Procedural memory is not all you need: Bridging cognitive gaps in llm-based agents. In *Adjunct Proceedings of the 33rd ACM Conference on User Modeling, Adaptation and Personalization*, UMAP Adjunct '25, page 360–364, New York, NY, USA. Association for Computing Machinery.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. 2024. Gated linear attention transformers with hardware-efficient training. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. 2025. Parallelizing linear transformers with the delta rule over sequence length. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Shibo Yu, Mohammad Goudarzi, and Adel Nadjaran Toosi. 2025. Efficient routing of inference requests across llm instances in cloud-edge computing.

Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Computation*, 31(7):1235–1270.

M. Zaheer, Guru Guruganesh, Kumar Avinava Dubey, J. Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. *ArXiv*.

Luca Zancato, Arjun Seshadri, Yonatan Dukler, Aditya Golatkar, Yantao Shen, Benjamin Bowman, Matthew Trager, Alessandro Achille, and Stefano Soatto. 2025. B'mojo: hybrid state space realizations of foundation models with eidetic and fading memory. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Shuangfei Zhai, Walter A. Talbott, Nitish Srivastava, Chen Huang, Hanlin Goh, Ruixiang Zhang, and J. Susskind. 2021. An attention free transformer. *ArXiv*.

Michael Zhang, Kush Bhatia, Hermann Kumbong, and Christopher Re. 2024. The hedgehog & the porcupine: Expressive linear attentions with softmax mimicry. In *The Twelfth International Conference on Learning Representations*.

Shen Zhuoran, Zhang Mingyuan, Zhao Haiyu, Yi Shuai, and Li Hongsheng. 2021. Efficient attention: Attention with linear complexities. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3530–3538.

A Appendix

A.1 Sourcing Methodology

Our survey followed a two-fold methodology: First, to determine which alternative model architectures to include, we began with a set of seed papers drawn from recent articles in the field, namely Wang et al. (2024c), Gu and Dao (2023), Sun et al. (2023), and Tay et al. (2022). From this base, we employed a backward and forward snowballing strategy: we examined the references cited within these seed papers (backward snowballing) as well as subsequent papers that cited them (forward snowballing). This iterative process enabled us to trace the development and recurrence of specific architectural primitives over time and across various research communities. Architectures that consistently reappeared in recent high-impact publications were included in the main body of our review. In contrast, those that were short-lived but had significant conceptual or empirical influence were included in Appendix A.2 as honorable mentions. Architectures with limited recurrence and marginal impact were excluded.

Second, for the chapter discussing the fundamental limitations of quadratic and sub-quadratic architectures, we conducted a systematic literature review. This involved querying several academic databases with the search term

("fundamental limitation") AND ("transformer" OR "attention" OR "subquadratic") AND ("natural language processing" OR "NLP" OR "language model")

to identify relevant theoretical and empirical work. The results, i.e., number of hits for each platform, and the search space (full text or abstract only), are stated in the following:

- ACL: 300 (full text)
- Semantic Scholar: 258 (full text)
- Google Scholar: 4430 (full text)*
- IEEE: 4 (abstract)

We then condensed our findings and reported on the very core of limitations that the other findings build upon. Secondary limitations were moved to Appendix A.3. *For Google Scholar, we used additional filtering to address the high number of hits and relatively low overall relevance. Cutoff for the SLR was 2025-06-18, but we continued to include individual relevant papers until paper submission.

A.2 Honorable Mentions

In our work, we have encountered various interesting and previously impactful subquadratic architectures, which, however, we were not able to include in the main body of this paper. This was usually due to a combination of limited space and our findings that these architectures were outperformed by others before they became relevant in the long run. For completeness, this section gives a brief overview of these works.

- **DeltaNet** Schlag et al. (2021) proposed DeltaNet, a linear transformer variant that retrieves and updates a value vector associated with each key using an update rule similar to the delta rule. DeltaNet employs a *diagonal plus low-rank* (DPLR) state-update mechanism similar to S4, enabling efficient parallelization across the temporal dimension and significantly improving training efficiency (Yang et al., 2025).
- **Hyena** Poli et al. (2023) introduced Hyena, a subquadratic alternative to attention. Hyena combines implicitly parameterized long convolutions with input-dependent gating mechanisms. Architecturally, Hyena resembles H3 (Fu et al., 2022) but substitutes the original S4 layer with global convolutions parameterized by multilayer perceptrons.
- **RetNet** Sun et al. (2023) introduced RetNet, a retention mechanism for sequence modeling that supports three computation modes: parallel (enabling efficient training), recurrent (providing low-cost $\mathcal{O}(1)$ inference, reducing latency and memory usage without sacrificing performance), and chunkwise recurrent (combining parallel encoding within chunks and recurrent summarization for efficient linear-complexity modeling of long sequences). At release, RetNet demonstrated strong scaling, efficient parallel training, and cost-effective inference.
- **TransNormerLLM** (Qin et al., 2024a): Introduced *TransNormerLLMs* (TNLs), whose architecture is specifically designed for lightning attention, and has additional modifications regarding positional embedding, linear

attention acceleration, gating mechanism, and tensor normalization.

- **Gated Linear Attention** Yang et al. (2024): introduce the hardware-efficient algorithm FlashLinearAttention, which they then generalize with data-dependent gates and use to replace standard attention with in a Transformer to propose *Gated Linear Attention* (GLA). GLA Transformers are especially effective at length generalization.

A.3 Additional Limitations of Attention

Some important secondary limitations of attention had to be cut from the main body of the paper due to a lack of space. We will list them in the following.

- Hahn (2020) prove that pure attention Transformers cannot handle bracket matching, iterated negation, or non-counter-free regular languages on long inputs, nor emulate stacks or arbitrary finite-state automata (unless layers or heads scale with input length).
- Sanford et al. (2023) show that single-layer, multi-head Transformers require polynomially more heads or dimensions to solve certain triple detection tasks, and likely struggle with higher-order tasks like Match3 (Sanford et al., 2023) without hints or augmentation. However, most real-world sequence problems decompose into pairwise relationships, aligning well with transformer capabilities.
- Huang et al. (2025) propose a theoretical framework to investigate length generalization in causal transformers that use learnable absolute positional encodings. By introducing constraints on how positional information can be utilized, their framework allows them to derive results for multilayer models. They formally prove problems with poor length generalization, such as copying sequences containing repeated strings. Although it remains an open question whether the expressivity of transformers goes beyond the complexity class TC^0, their findings suggest a potential distinction between problems solvable within TC^0 and those for which length generalization is feasible with absolute positional encodings.
- Amiri et al. (2025) investigate systematic lower bounds on the number of CoT steps required for various algorithmic problems within a hard-attention setting. Their analysis demonstrates that the required CoT length necessarily must scale with input length, thereby constraining the ability of self-attention models to solve these tasks efficiently with small inference-time compute.
- Peng et al. (2024) prove that a single transformer layer is not able to do function composition if the domain size of the functions is larger than the dimension parameters of the transformer. Moreover, they show that if we leverage CoT, the model needs to generate a $\Omega(\sqrt{n})$ long prompt to solve iterated function composition, with n being the number of tokens in the prompt. They assume that multi-layer transformers struggle as well.
- Saparov et al. (2025) argue that transformers with standard training will not have robust searching and planning abilities, no matter their number of parameters. For small graphs, a model with effectively limitless and idealized training data can learn to search. Nevertheless, according to them, even if a model can use search in-context (i.e., CoT), it still struggles with search on larger graphs.

A.4 Benchmarking Details

Model References Titans (Behrouz et al., 2024), Griffin (De et al., 2024), HGRN2 (Qin et al., 2024d), Mamba2 (Dao and Gu, 2024), xLSTM (Beck et al., 2024), BMoJo (Zancato et al., 2025), RWKV7 (Peng et al., 2025), Samba (Ren et al., 2025), Jamba (Lenz et al., 2025), Qwen2.5 (Qwen et al., 2025), Llama3.1 (Grattafiori et al., 2024), Mixtral (Jiang et al., 2024)

Benchmarks (accuracy based) MMLU (Hendrycks et al., 2020), Lambada (Paperno et al., 2016), PIQA (Bisk et al., 2020), BBH (Suzgun et al., 2023), ARC-E and ARC-C (Clark et al., 2018), Winogrande (Sakaguchi et al., 2021), HellaSwag (Zellers et al., 2019), GSM8k (Cobbe et al., 2021), and HumanEval (Chen et al., 2021)

Result Sourcing We do not have the computational resources to run our own evaluations for all models on all benchmarks. Instead, we chose to use the results from Qwen et al. (2025) for Qwen2.5 and Llama 3.1, Peng et al. (2025) for Llama 3.2 and RWKV, due to their consistent evaluation suites. For all other models, we gathered the results from

their original technical papers, ensuring consistency to the best of our knowledge. Nevertheless, some inconsistencies, namely in the number and type of tokens used during training, and differences in the number of shots for some task/model combinations, remain.