

Preserving Memory Function in Hybrid Language Models: A Survey of State, Evidence, Access, and Granularity

Zhuangfei Hu¹ Runqing Xu¹ Yuchen Li² Yongqi Zhang^{1,*}

¹The Hong Kong University of Science and Technology (Guangzhou)

²Baidu Inc., Beijing, China

zhu012@connect.hkust-gz.edu.cn, yongqizhang@hkust-gz.edu.cn

Abstract

Hybrid language models combine Transformer attention with recurrent, state-space, or online-updated components to lower long-context serving cost while preserving memory fidelity. Existing surveys mostly organize this space by architecture family, chronology, or efficiency technique, leaving workload-specific memory functions underspecified. We propose State-Evidence-Access (SEA) for this constraint: bounded State may lose distinctions needed by future queries not fixed in advance. Architectures therefore allocate active State, addressable Evidence, and a per-step Access budget to jointly support reusable structure, query-conditioned recovery, and effective use. This view helps explain why attention ratios or average perplexity can be misleading: a hybrid can save memory while losing the retrieval, state-update, or access function needed by the task. We survey hybrids by allocation locus—from key–value (K–V) and head-level mechanisms through layer, block, external-memory, and system-cache policies—and characterize their *function-granularity mismatch* risks. For workloads such as long-document QA, SEA attributes failure to missing evidence, unstable state updates, access bottlenecks, or poor integration.

1 Introduction

Long-context language models must retain relevant evidence at manageable inference cost. Transformer self-attention (Vaswani et al., 2017) keeps a key–value (K–V) store for direct token access, but KV-cache memory grows with length. State-space models (SSMs) and gated RNNs compress history into bounded active state (Gu and Dao, 2023), enabling streaming generation but weakening arbitrary copying and retrieval (Jelassi et al., 2024; Wen et al., 2025). State tracking adds pressure because

common SSMs can share bounded-depth Transformer limits for reliable state evolution (Merrill et al., 2024). Hybridization is therefore not merely an efficiency compromise: no standard mechanism cleanly provides addressable token evidence, reliable compressed-state evolution, and bounded cost for accessing and updating memory.

Recent hybrids combine attention with recurrent, state-space, or online-updated components. Jamba (Lieber et al., 2024), Griffin (De et al., 2024), Zamba (Glorioso et al., 2024), Hymba (Dong et al., 2025), Nemotron-H (NVIDIA, 2025), and Falcon-H1 (Zuo et al., 2025) span these regimes. Prior surveys catalogue SSM and Mamba families (Qu et al., 2024), long-context techniques (Liu et al., 2025; Wang et al., 2024b), efficient attention (Sun et al., 2025c), augmentation (Mialon et al., 2023), and KV-cache management (Li et al., 2025). Our focus is complementary (Table 1): we ask which retrieval, update, or access function survives attention replacement, sparsification, or compression under separate State, Evidence, and Access budgets. Appendix H gives the detailed comparison. Bae et al. (2025) empirically search Transformer–Mamba ratios and placements in tested stacks. SEA complements this work by synthesizing across architectures to ask which resource, function, allocation locus, and mismatch explain a result.

Hybrid language models are *function-preserving memory-allocation systems*: a future query may require a distinction lost by bounded compression. SEA asks what reusable structure State can carry, what Evidence must remain addressable until its relevance is known, and whether the Access budget can retrieve and integrate it. A mismatch arises when compressed State replaces addressable Evidence, retained Evidence cannot be read or integrated under budget, or capacity is assigned at the wrong granularity.

After identifying the resource, SEA locates the allocation decision. This decision may occur at a

*Corresponding author.

Neighboring line	Primary organizing emphasis
SSM and Mamba	Mechanism families and applications.
Long-context	Context-extension techniques and benchmarks.
Efficient attention and KV cache	Attention cost and serving memory.
Augmented LMs	External tools and retrieval augmentation.
Hybrid design study	Empirical ratio and placement search in tested stacks (Bae et al., 2025).
This survey	Memory function and allocation locus under SEA budgets.

Table 1: Positioning relative to neighboring literature lines.

K-V pair or head, at a layer or block, in an external store, during training-time conversion, or in a serving-time cache policy. Allocation granularity therefore states where capacity is spent; mechanistic probes test whether that choice implements the required function.

At the finest granularity, this becomes a K-V allocation question: what evidence remains addressable, and what is compressed, externalized, or evicted? In long-document QA, SEA localizes an error to the resource or integration step that failed. Section 4.1 formalizes this question; Appendix B records evidence use. In summary, our contributions are:

- **SEA framework.** We organize hybrid memory design around State, Evidence, and Access budgets, with memory functions as roles over those resources.
- **Multi-axis taxonomy.** We recast hybrids by resource role, mechanism, allocation granularity, and intervention phase, then use representative design patterns to track mismatch risks.
- **Function-preservation tests.** We provide a five-step comparison guide and a controlled intervention that connect capability pressures to resource requirements, K-V allocation, and alignment objectives.

Organization. Section 2 defines the scope and survey protocol. Section 3 starts from capability pressure and turns Figure 1 into a five-step practitioner workflow. Section 4 then maps required functions to SEA resources and the fine-grained K-V lens. Section 5 locates hybrid allocation choices, and Section 6 asks what function must survive replacement or distillation. Section 7 presents open problems, and Section 8 concludes the survey.

2 Scope, Definitions, and Survey Methodology

Scope and definitions. This survey covers autoregressive hybrids: language models that combine attention with recurrent, state-space, linear-attention, online-updated, external-memory, or cache-policy components to change long-context memory allocation. A model is *hybrid* when multiple memory mechanisms or allocation policies are architecturally meaningful, not merely because it uses an optimized attention kernel. We use *SSM* broadly for structured recurrent state updates. *Memory function* is the workload role of a resource, such as lookup, state update, associative prediction, external access, or cache control. *Allocation locus* is the unit assigned that role: a K-V pair, head, layer, block, external store, or cache policy. *Function-granularity mismatch* means that this locus is too coarse or misaligned.

Inclusion boundaries. We include theory, mechanisms, benchmarks, conversion methods, and systems work that clarify retained evidence, compressed state, online associative update, external lookup, or cache and access policies for autoregressive text modeling. Work centered on multimodal, vision, graph, genomics, or time-series hybrids is included only when it clarifies a boundary case for language-model memory allocation.

Survey protocol. We conducted a targeted survey using ACL Anthology, arXiv, OpenReview, Semantic Scholar, and major conference and systems proceedings for 2017–2026 work using keyword groups around *hybrid language model*, *SSM and Mamba*, *linear and recurrent language model*, *long context*, *memory compression*, *retrieval head*, *KV cache*, *external memory*, and *Transformer-to-SSM or Transformer-to-Mamba distillation*, then snowballed from related surveys and benchmarks. Each work is coded by resource role, mechanism, allocation granularity, intervention phase, memory function, mismatch risk, benchmark or probe type, publication status (peer-reviewed, preprint, technical report, or blog), and evidence type (theory, benchmark, or mechanistic analysis). The appendices document the protocol and how each source is used.

3 Capability Pressure and the SEA Workflow

SEA starts from the workload and externally observable behavior rather than from an architecture label. A reader first asks what behavior the task requires, then maps that function to State, Evidence, or Access, locates the allocation decision, and tests the resulting mismatch. This order prevents an attention ratio or cache size from standing in for the function that the retained capacity must support. Figure 1 summarizes the path.

3.1 Workload Pressure Clusters

Context-access tasks require recovery of a specific token, binding, or pattern from the prefix. Copying lower bounds (Jelassi et al., 2024), in-context retrieval analysis (Wen et al., 2025), and broader subquadratic limitations (Alman and Yu, 2025) show why some workloads need direct access to retained token evidence rather than only bounded compressed state. Context-dependent sparse attention, as in HAX and CDSA (Zhan et al., 2025), illustrates the corresponding design requirement: select evidence for the current query instead of relying on a fixed sparse pattern. Mamba-like models can still exhibit ICL behavior in smooth or low-dimensional regimes (Grazzi et al., 2024; Park et al., 2024); exact copying, associative recall, and retrieval-heavy ICL are the sharper stress tests.

State-evolution tasks instead require reliable latent updates across operations. Common SSMs, linear recurrent variants, or constrained state-space parameterizations can impose update-rule limits distinct from retrieval failure (Merrill et al., 2024; Grazi et al., 2025). Streaming and long-context serving add a third pressure: preserve usable history while controlling storage, selection, routing, and reading costs. These three clusters correspond to addressable lookup, stable compressed updates, and budgeted access to stored memory.

Long-document QA makes the separation concrete. A question can require distant answer-span lookup and provenance, maintenance of document-level constraints, and enough retrieval and integration capacity to use the stored evidence. Calling all three a “long-context failure” obscures which resource or integration pathway should be tested.

3.2 A Five-Step Practitioner Guide

Table 2 turns the workflow into a comparison procedure. The example holds the long-document QA

workload fixed and compares RecurrentGemma (Botev et al., 2024) with Hymba (Dong et al., 2025). The entries are SEA-predicted, testable hypotheses derived from each allocation pattern unless the cited source reports the result.

4 SEA Resource Budgets and the Fine-Grained K-V Lens

Step 3 of the practitioner workflow maps a required function to the resource that must support it. *State* is the bounded latent representation active in recurrent or online-updated dynamics. *Evidence* is retained material that remains individually addressable, either in a K-V cache or an external store. *Access* is the per-step budget for making stored memory usable. It covers three distinct operations: selection and routing, reading and integration, and writing and updating.

We write B_t for active-State capacity, S_t/M_t for retained or external addressable Evidence, and R_t for the logical Access budget. Examples of an enforceable R_t cap are selected tokens, K-V reads, retrieval calls, or top- k . These caps are hardware-independent and directly enforceable; bytes moved and latency are hardware- and software-dependent costs that must be reported separately. Table 3 compares the resulting regimes, and Section 4.1 makes their allocation variables explicit.

At object level, Transformer attention and its K-V cache keep past token positions addressable by content query (Vaswani et al., 2017), whereas SSM hidden state represents patterns, transitions, and predictive relations in bounded latent state (Gu et al., 2022; Gu and Dao, 2023). A *K-V pair* is one token position’s key and value vectors in an attention layer. Retaining the pair keeps that position individually addressable rather than only available through compression.

Resource roles are distinct from mechanisms. Recurrent, SSM, and fast-weight variables can all realize State (Gu and Dao, 2023; Schlag et al., 2021); attention, local caches, and external stores can realize Evidence. Externalizing more Evidence does not enlarge active State and does not make the Evidence free to use: selection, retrieval, and integration still consume Access.

Resource roles and lineage. The regime table condenses a larger taxonomy documented in Appendix C. Transformer-XL and Compressive Transformer place segment or compressed memories between retained Evidence and State (Dai et al., 2019;

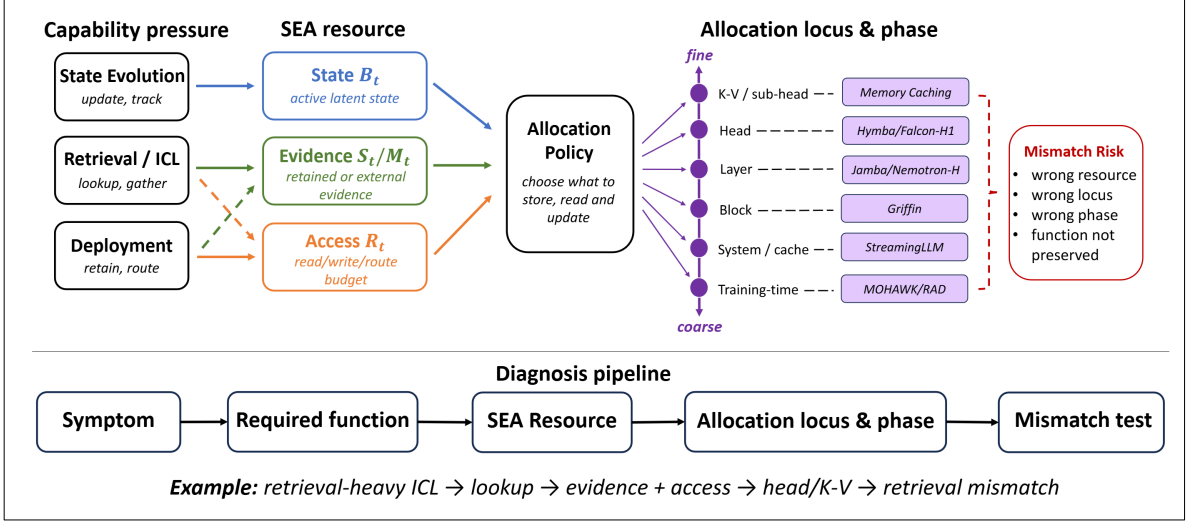


Figure 1: Five-step workflow for function-preserving allocation: fix a workload or symptom, identify its required memory function, map that function to SEA resources, locate where and when capacity is allocated, and test the predicted mismatch. Solid arrows mark dominant pressures; dashed arrows mark secondary couplings.

Rae et al., 2020); kNN-LM, RETRO, and Memorizing Transformers externalize Evidence (Khandelwal et al., 2020; Borgeaud et al., 2022; Wu et al., 2022). RMT, Landmark Attention, LongNet, Hyena, and Mega change memory tokens, selection patterns, or sequence operators (Bulatov et al., 2022; Mohtashami and Jaggi, 2023; Ding et al., 2023; Poli et al., 2023; Ma et al., 2023). These families are boundary evidence for the same resource distinction, not additional SEA resources.

Explicit attention provides content-addressable Evidence, while sparse or local attention adds a selection policy (Vaswani et al., 2017; Deng et al., 2024). Structured SSMs, gated RNNs, RWKV, and RetNet instead expose history through compressed State updates (Gu et al., 2022; Gu and Dao, 2023; Peng et al., 2023; Sun et al., 2023); copying and retrieval probes test what that State does not preserve (Jelassi et al., 2024; Wen et al., 2025).

Online-updated and external roles. Linear attention, GLA, DeltaNet, TTT, feature-map approximations, and product-key fast weights make State associative or predictive (Katharopoulos et al., 2020; Yang et al., 2024, 2025; Sun et al., 2025b; Zhang et al., 2024a; Zhao and Jones, 2026). RAG, product-key memory, and Engram instead add external or deterministic lookup Evidence (Mialon et al., 2023; Zhao and Jones, 2026; Cheng et al., 2026). Structured State Space Duality places attention and SSMs in a shared algebraic space (Dao and Gu, 2024), but matrix structure, State capacity, and Access still determine their behavior.

4.1 The Fine-Grained K-V Allocation Lens

The resource-budget distinction raises a sharper question: at K-V or sub-head granularity, which evidence remains addressable, compressed into state, externalized, or ignored? We use a compact formal lens to define measurable intervention targets: task-induced query distributions, evidence units, resource budgets, access policies, and failure attribution.

Let the processed K-V evidence at time t be

$$H_t = \{(k_i, v_i)\}_{i \leq t}, \quad q \sim \mathcal{Q}_t,$$

where q is a future query drawn from the task-induced query distribution. The model has three resources: an active latent state $h_t \in \mathbb{R}^{B_t}$, a K-V retention cache

$$C_t = \{(k_i, v_i) : \pi_i = 1\}, \quad |C_t| \leq S_t,$$

and an external or persistent memory M_t . Let $\mathcal{R}_t = (h_t, C_t, M_t)$ denote the available resources. The allocation decisions are K-V retention indicators $\pi_i \in \{0, 1\}$, compressed or external updates u_i , and a routing policy $\rho(q)$ that decides which resources a query may read.

For a full-attention target, define

$$r^*(q, H_t) = \sum_{i \leq t} \alpha_i(q, H_t) v_i, \\ \hat{r}_\rho(q; \mathcal{R}_t) = A_{\rho(q)}(q; \mathcal{R}_t),$$

where $A_{\rho(q)}$ is the retrieval procedure induced by the chosen route. The budgeted allocation problem

Step	Input and question	Required output	RecurrentGemma	Hymba
1. Workload	Fixed task, context length, and symptom: what must the model do with earlier context?	Specific pressure, not “long context” generically.	QA whose answer or bridge evidence lies beyond the local window.	The same QA instances and context length.
2. Function	Annotated answer and bridge spans: is success driven by lookup, state evolution, or integration?	Primary and secondary memory functions.	Distant content-addressable lookup is primary; document-state tracking is secondary.	The same distant-lookup function is primary; document-state tracking is secondary.
3. Resource	Which SEA resource must carry the function?	State, Evidence, or Access requirement with a proxy.	Recent K-V Evidence remains addressable; the far tail is represented in recurrent State.	Attention heads retain addressable K-V Evidence alongside recurrent State.
4. Locus	Where and when is capacity assigned?	Allocation map at K-V, head, layer, block, store, or cache-policy level.	A block-level split between a local-attention window and global recurrence.	A head-level split between attention and recurrent channels inside a hybrid block.
5. Test	Which controlled intervention distinguishes the candidate mismatch?	Falsifiable risk, intervention, and metric.	Protect or evict distant gold spans and vary the window; measure retention, retrieval, and answer accuracy.	Sweep attention-head capacity or cache capacity separately; measure retrieval recall and answer accuracy under the same workload.

Table 2: Five-step SEA comparison guide. Each step has an input, question, and output; the final columns apply the same step to a fixed long-document QA setting.

Regime	State B_t	Evidence S_t/M_t	Access R_t	Consequence
Transformer	$O(1)$ query	$S_t = O(n)$	$O(n)$ read	Exact evidence; growing cache and read cost.
Fixed SSM or RNN	$O(1)$	$S_t = 0$	$O(1)$	Cheap streaming; compressed history.
Linear attention, TTT, or fast weights	predictor or matrix	implicit	$O(1)$ update	Adaptive state; bounded capacity.
Local attention with recurrence	$O(1)$ global	$S_t = O(w)$	$O(w) + O(1)$	Exact window; compressed far tail.
Head- or layer-level hybrid	mixed	selected S_t	structured	Attention must match function.
External lookup	small	$M_t \gg S_t$	retrieval-limited	Scaled storage; retrieval bottleneck.
Growing or checkpointed state	$O(g(n))$	optional S_t	adaptive	Between recurrence and full KV.

Table 3: Resource regimes in autoregressive inference.

is then

$$\begin{aligned}
& \min_{\pi, u, \rho} \mathbb{E}_{q \sim \mathcal{Q}_t} [\ell(r^*(q, H_t), \hat{r}_\rho(q; \mathcal{R}_t))] \\
& \text{s.t. } |C_t| \leq S_t, \dim(h_t) \leq B_t, \text{cost}(\rho, u) \leq R_t.
\end{aligned}$$

The loss may measure attention-output recovery, target-value recovery, or downstream prediction error; the point is to expose the tradeoff between active-State capacity, addressable Evidence, and the Access budget. In practice, $\text{cost}(\rho, u)$ can count cache reads, routing or eviction calls, and external lookups. Bytes moved and latency are reported alongside those logical counts for a specified stack. Table 4 gives stack-independent proxies for these variables alongside the physical costs that require a named implementation. The full-attention target provides a reference for addressable evidence. State-evolution tasks may require task losses or state-transition objectives instead.

Operationally, start from a workload-induced query distribution $\mathcal{Q}_t$, choose a probe or intervention, estimate which K-V entries, heads, or layers are function-sensitive, then test whether failure comes from missing evidence, state-update loss, access limits, or poor integration. Copying and associative recall point to retained evidence; state tracking to compressed-state updates; and long-context serving to cache, routing, and integration policies.

Common architectures become special cases. A full Transformer uses $S_t = O(t)$, $B_t = O(1)$, and dense $R_t = O(t)$ access; a fixed SSM or RNN sets $S_t = 0$, $B_t = O(1)$, and $R_t = O(1)$; sparse attention selects an evidence subset; external memory enlarges M_t but remains bounded by R_t ; and growing-latent-state regimes scale $B_t = O(g(t))$ with $1 \ll g(t) \ll t$. Memory Caching (Behrouz

Quantity	Workload proxy	Reportable unit or cap	Stack dependence
Q_t	Empirical distribution of task queries at the fixed context length; for QA, the evaluation questions.	Number and strata of queries; answer and bridge-evidence annotations.	Workload-defined and hardware-independent.
B_t	Active recurrent-State slots or latent variables available when the query arrives.	Slots, dimension, token-equivalent capacity; bytes with dtype stated.	Logical shape is independent; bytes depend on representation and implementation.
S_t/M_t	Retained K-V Evidence, retained gold spans, or externally stored records.	K-V entries or tokens, gold-span retention rate, records; bytes with layer and data type stated.	Counts are independent; byte cost depends on model and cache format.
R_t	Per-step cap on selecting and routing, reading and integrating, or writing and updating memory.	Selected tokens, K-V reads, retrieval calls, or top- k ; also bytes moved and latency.	Logical caps are directly enforceable; traffic and latency are stack-dependent.

Table 4: Practical proxies for the K-V lens. Logical quantities are comparable across stacks; physical costs must name the implementation.

et al., 2026) caches recurrent checkpoints and merges them with online state, illustrating state growth without literal token-level K-V selection. Hybrid Associative Memories (Lufkin et al., 2026) and fast-weight product-key memory (Zhao and Jones, 2026) provide related fixed-allocation designs, leaving adaptive allocation open.

Worked examples beyond the active K-V context. δ -mem adds a compact online associative State pathway to a frozen attention backbone; in the reported setting, an 8×8 state is updated by a delta rule and its readout steers attention through low-rank corrections (Lei et al., 2026). This adds active-State capacity and changes State construction, but the matrix is not a set of arbitrary past token records that remain individually addressable. Sleep-like consolidation instead performs additional recurrent passes before hard K-V eviction while keeping the State representation and single-pass wake-time prediction path fixed (Lee et al., 2026). It therefore adds State-construction compute, not B_t or addressable Evidence; after eviction, discarded K-V pairs still cannot be revisited directly.

Four-way controlled diagnostic. For the running workload in Table 2, fix the model, task instances, context length, decoding, and logical resource caps. Annotate answer spans and bridge Evidence before intervening on one pathway at a time:

1. **Evidence:** protect or evict annotated spans while holding State and Access fixed; report gold-Evidence retention and answer accuracy.
2. **State:** change active-State capacity or the compression or update path without adding addressable K-V Evidence; report a state-sensitive probe and answer accuracy.
3. **Access:** sweep selected-token, K-V-read, or

retrieval-call caps over retained Evidence; report retrieval recall, answer accuracy, and the logical R_t used.

4. **Integration:** supply oracle-retrieved Evidence under the same storage setting; report answer accuracy conditional on successful retrieval.

Here Integration isolates the use component of Access after retrieval succeeds; it is a diagnostic control, not a fourth resource. The intervention that restores or degrades the outcome supports the attribution. A single accuracy drop without these controls does not.

5 Architecture Spectrum: Multi-Axis Hybrid Design

With the required function and SEA resource fixed, Step 4 locates the allocation decision. Hybrid models assign capacity at granularities from K-V operations and heads through layers, blocks, external stores, and cache policies. The patterns below state where future-sensitive Evidence remains addressable, becomes compressed State, or is subject to an Access policy. Table 6 summarizes representative systems by this allocation handle and separates reported evidence from SEA-predicted risks. Table 2 applies the same map to RecurrentGemma and Hymba.

Four axes keep different design choices separate (Table 5). Resource role asks what supports the function; mechanism asks how history is represented or read; granularity asks where capacity is assigned; and phase asks when the choice is imposed. Appendix E gives the larger architecture catalogue.

K-V and sub-head granularity. At the finest granularity, the decision unit is the K-V pair or sub-head channel: which evidence remains addressable

Axis	Values and test question
Resource role	Active State; addressable Evidence; Access budget. <i>Question:</i> what preserves the function?
Mechanism	Attention; recurrence or SSM; linear attention or fast weights; external memory; cache policy. <i>Question:</i> how is history stored or read?
Granularity	K-V or sub-head; head; layer; block or module; external store; system policy. <i>Question:</i> is the unit fine enough?
Phase	Pretraining; conversion or distillation; inference routing; serving-time cache management. <i>Question:</i> when is the choice imposed?

Table 5: Four axes for hybrid memory allocation.

rather than summarized? Current hybrids mostly allocate at coarser head, layer, or block loci, but Memory Caching, hybrid associative memory, and fast-weight product-key memory point toward finer choices (Behrouz et al., 2026; Lufkin et al., 2026; Zhao and Jones, 2026). These are not literal K-V-pair solvers: cached checkpoints or associative updates may contain useful history yet omit the evidence a later query needs.

Layer-wise interleaving. The common strategy inserts full-attention anchors into a mostly recurrent stack so accumulated context-access demands can be resolved and passed downstream. Jamba (Lieber et al., 2024), Samba (Ren et al., 2025), and Nemotron-H (NVIDIA, 2025) use different mixes of attention, recurrence, and gated linear attention. Because attention ratio and placement affect capabilities (Bae et al., 2025), and ICL functions can concentrate unevenly (Wang et al., 2025b), placement is the allocation variable, not ratio alone.

Intra-layer and head-level fusion. A finer strategy assigns heads or channels inside a block to different memory mechanisms. Hymba (Dong et al., 2025) and Falcon-H1 (Zuo et al., 2025) mix attention-like and recurrent channels; mechanistic work suggests retrieval-critical heads can preserve retained evidence while other heads carry compressed dynamics or local refinement (Bick et al., 2025b; Michalak and Abreu, 2025). The split helps only when protected heads align with retrieval- or update-critical functions.

Shared attention modules. Shared attention routes multiple positions through one centralized retrieval module. Zamba (Glorioso et al., 2024) lowers KV-cache overhead this way, but one module may not serve every layer-local retrieval need.

Local attention plus global recurrence. Local attention plus global recurrence gives direct access to a short recent window and compresses the tail into recurrent state. Griffin (De et al., 2024) and RecurrentGemma (Botev et al., 2024) instantiate this division. The design is attractive for state evolution but can be fragile if far-past evidence becomes retrieval-critical and cannot be reconstructed from state.

Scheduled switching and query-conditioned access. Under the decision in Section 4.1, allocation can vary by token position, layer, or query, but these are distinct mechanisms. TransMamba (Li et al., 2026) uses predefined, layer-specific transition points to switch from attention to SSM computation as token position increases. TransXSSM (Wu et al., 2025a) instead uses a static periodic pattern of attention inside an SSM-dominant stack. HAX (Zhan et al., 2025) selects sparse retained evidence conditioned on query-key context within a fixed hybrid block. None of these methods directly predicts future retrieval relevance. Content-dependent module allocation remains open: a controller must act before future relevance is fully observable. Routing quality and training stability also depend on which representations and pairwise relations are available to the controller (Basu, 2026).

Training-time allocation and conversion. Distillation and conversion *compile* a teacher’s retrieval behavior into recurrent, linear-attention, or hybrid weights. We treat them as intervention-phase choices: the risk is preserving average predictions while losing teacher components that carry retrieval, state-update, or access behavior.

System-level and cache allocation. At deployment time, KV capacity, recurrent-state size, external evidence, the Access budget, and cache policy decide which memory functions remain usable per token. StreamingLLM retains attention sinks (Xiao et al., 2024b); InfLLM retrieves evicted blocks (Xiao et al., 2024a); DuoAttention uses head-aware budgets (Xiao et al., 2025); FlashAttention changes the physical cost of dense access (Dao et al., 2022; Dao, 2024); and vLLM, H2O, SnapKV, and KIVI provide paging, eviction, selection, or quantization (Kwon et al., 2023; Zhang et al., 2023; Li et al., 2024a; Liu et al., 2024). SideQuest (Kariyappa and Suh, 2026) illustrates the failure mode: cache savings are valid only if multi-hop reasoning evidence remains readable.

System line	Allocation handle	Pub.	Evidence basis; SEA inference
Jamba; Jamba-1.5	Layer anchors in Mamba-heavy stack	pre.	SEA: misplaced-retrieval risk (Lieber et al., 2024; Jamba Team, 2024)
Griffin; RecurrentGemma	Local KV and recurrent global state	pre.	SEA: risk of far-tail Evidence loss (De et al., 2024; Botev et al., 2024)
Zamba	Shared attention; amortized KV	pre.	SEA: shared-module coverage gap (Glorioso et al., 2024)
Hymba; Falcon-H1	Head- and block-level fusion; attention plus recurrent channels	mixed	SEA: head-to-function gap (Dong et al., 2025; Zuo et al., 2025)
Samba; Nemotron-H	Production mix; attention, recurrence, and linear attention	mixed	SEA: placement-to-function gap (Ren et al., 2025; NVIDIA, 2025)
Attn.–Mamba conversion	Replace or linearize attention	mixed	BOTH: output alignment can miss retrieval function (Bick et al., 2024, 2026, 2025a)
StreamingLLM; InfLLM	Cache sinks plus recent tokens; evicted-block retrieval	pub.	SEA: risk of an evicted-Evidence miss (Xiao et al., 2024b,a)
DuoAttention; SnapKV	Head- and prompt-token KV selection	pub.	BOTH: cache-budget accuracy trade-off; future-relevance risk (Xiao et al., 2025; Li et al., 2024a)
vLLM; H2O; KIVI	Paging; eviction; KV quantization	pub.	SEA: byte-to-function gap (Kwon et al., 2023; Zhang et al., 2023; Liu et al., 2024)

Table 6: Representative systems by allocation handle, publication status, and evidence basis. REP.=measured result in the cited source; SEA=mismatch risk inferred from the allocation; BOTH=reported result plus SEA inference (pub.=published; pre.=preprint or report; mixed=published and preprint records).

6 Function-Granularity Mismatch: What Must Be Preserved?

Step 5 asks whether the locus identified in Section 5 preserves the required function. A model can preserve perplexity while losing copying, associative recall, or retrieval-heavy ICL if remaining attention misses retrieval-carrying components. G&A heads are the clearest mechanistic case below. The four-way procedure in Section 4.1 keeps these alternative explanations separate before a mechanistic result is attributed to retrieval loss.

Three mismatch modes. Retrieval mismatch asks if needed evidence remains addressable; state-update mismatch if compressed dynamics maintain task-relevant latent variables; and access-policy mismatch if retained or external evidence remains readable under budget. Retrieval receives detail because evidence is strongest at head and layer loci; the other two remain parallel tests for SSM updates and cache policies.

Retrieval-critical components. Attention heads specialize: Bick et al. (2025b) identify *Gather-and-Aggregate* (G&A) heads that gather and route relevant signals, while Wu et al. (2025b) characterize retrieval-specialized heads by factuality and routing. Unlike local-pattern, nearby-aggregation, or positional-anchor heads, these implement addressable-evidence access. Replacing,

distilling, or caching across them can preserve budget or perplexity yet lose the retrieval function (Section 3). Parallel failures arise when latent state is not maintained or stored evidence becomes unreadable.

Function vs. aggregate quality. Aggregate language-modeling quality does not identify which components carry retrieval. G&A analyses instead find that retrieval-critical behavior is concentrated in a few heads (Bick et al., 2025b). ICL and Transformer–SSM evaluations (Wang et al., 2025b; Arora et al., 2025) likewise show that aggregate quality can hide different routes to context use. Replacing attention should retain G&A-type retrieval or supply an explicit mechanism.

Alignment targets for distillation. Distillation and conversion choose which teacher signals to compile into cheaper recurrent, linear-attention, or hybrid weights. MOHAWK (Bick et al., 2024), Goldstein et al. (2025), Mamba-in-Llama (Wang et al., 2024a), LoLCATs (Zhang et al., 2025), Attention Bridge (Wang et al., 2025a), and Priming (Chattopadhyay et al., 2026) illustrate weight reuse, attention linearization, representation alignment, and layer transfer. Priming, for example, replaces selected attention layers with SSM layers, transfers source weights, and aligns each new layer before end-to-end adaptation. Logit or hidden-state

Task, metric, and baseline	Reported result	Scope caveat
<i>Bae et al. (2025)</i> Allocation setting: 1B inter-layer Transformer:Mamba; 60B tokens; ratios 1:1 and 1:5		
DCLM/PG19 NLL; five-task average. Baseline: 1:0 Transformer; 0:1 Mamba.	1:1: NLL 2.725/2.847, average 54.0; 1:5: NLL 2.735/2.861, average 53.3; authors select about 1:5.	Tested grid only. At 350M, 1:3 leads (50.2). At matched ratios, front is consistently worse; middle leads only at 1:12; 1B/1:5 end placement leads the few-shot average.
<i>Michalak and Abreu (2025)</i> Allocation setting: Jamba-Mini-1.6; top- k prefill; $k = 5/32$ heads ($\approx 15\%$)		
NIAH retrieval; MMLU five-shot. Baseline: Jamba base; fully ablated $k = 0$.	$k = 5$ is the lowest tested near-perfect NIAH setting; MMLU drops six points, retaining 84% relative to base.	NIAH-selected and model-specific; 5/32 is not a global layer or K-V fraction, and 84% is retention, not accuracy.
<i>DuoAttention (Xiao et al., 2025)</i> Allocation setting: Llama-2/3 MHA or GQA; head-aware K-V budgets; A100 BF16		
NIAH/LongBench; memory; decoding speed. Baseline: Full attention; matched-budget K-V compression.	Maximum MHA: 2.55 $\times$ memory reduction and 2.18 $\times$ decoding speedup; 25% MHA and 50% GQA budgets are comparable on NIAH and most LongBench tasks.	Maxima vary by model, attention type, hardware, context, and budget; “comparable” is not zero loss on every task.

Abbreviations: NLL, negative log-likelihood; NIAH, needle-in-a-haystack; MHA, multi-head attention; GQA, grouped-query attention; BF16, bfloat16. DCLM, PG19, MMLU, and LongBench are corpus or benchmark names.

Table 7: Source-scoped attention-allocation results; values are not pooled into a universal “minimal attention” percentage.

alignment alone need not preserve retrieval, state, or access roles: RAD (Bick et al., 2026) targets context-access behavior directly, whereas Llama (Bick et al., 2025a) enables cross-architecture distillation without explicitly preserving G&A heads.

How much attention is needed? Table 7 reports the source-scoped ratios, placement result, and caveats behind this question. The practical question is how many layers or retrieval heads preserve context access. A few specialized components can carry disproportionate retrieval behavior (Michalak and Abreu, 2025), and targeted full attention can protect retrieval when many layers are linearized (Goldstein et al., 2025). Attention should therefore be allocated to identified retrieval-carrying components, not by a fixed ratio.

7 Open Problems

Function-aware and adaptive allocation. The K-V objective in Section 4.1 should yield head-, layer-, block-, and system-level policies for what stays addressable, is compressed, or adapts. SSD (Dao and Gu, 2024) unifies attention and SSMs algebraically, but not by allocation granularity; Memory Caching (Behrouz et al., 2026) is an early boundary case. Conversion and distillation need functional targets. Controllers may combine $O(1)$ recurrence, sublinear active state, and sparse retained evidence; when can this replace $O(n)$ access without losing independent bindings?

Evaluation and reporting. Benchmarks merge copying, associative recall, retrieval-heavy ICL, state tracking, and long-context lookup. MQAR

in Zoology (Arora et al., 2024a), copying lower-bounds (Jelassi et al., 2024), LongBench (Bai et al., 2023), and SSM and hybrid evaluations (Mittra et al., 2025) separate pieces. Protocols should map failures to SEA pressures and allocation loci, including multilingual and safety-critical settings where tokenization, provenance, or sensitive data change required addressability.

Systems co-design. KV paging, eviction, quantization, sparse kernels, and recurrent layouts constrain R_t and S_t . Report memory-function trade-offs with throughput and footprint so systems gains remain separable from retrieval or state-update loss.

8 Conclusion

Before future queries are known, bounded compression may discard distinctions that later matter. SEA tests this risk at the allocation locus by measuring preservation, recovery, and effective use.

Limitations

This survey is a targeted synthesis of autoregressive language-model memory mechanisms, not a full systematic review or exhaustive model catalogue. Vision, audio, and multimodal hybrids are treated only as boundary cases. Several recent systems and benchmark analyses are preprints or technical reports; we use them as evidence for emerging patterns rather than as conclusive quantitative support. Formal and mechanistic claims are interpreted under their source assumptions.

Potential risks. This survey introduces no new model, dataset, or deployment system, so its direct risks are limited. Indirectly, more efficient long-context and hybrid models may lower the cost of large-scale generation, retrieval, surveillance-scale document analysis, and agentic tool use. Persistent memory, external retrieval, and KV retention may also keep sensitive evidence available longer than users expect, while cache compression or eviction can remove safety-relevant instructions, constraints, provenance, or consent information. Deployment therefore requires task-specific safety, privacy, and misuse evaluation.

Exact historical search dates, paper-level exclusion reasons, and independent coding by multiple reviewers are unavailable for this survey. Long-context benchmarks also remain English-heavy and often synthetic, with limited coverage of multilingual or low-resource memory behavior. “Retrieval” here denotes content-addressable access in memory systems. Evaluating information-retrieval system performance is outside our scope.

Acknowledgments

This work was supported by Guangdong Basic and Applied Basic Research Foundation Project 2025A1515010304, Guangdong Province Project 2024QN11X088, and Guangzhou Science and Technology Planning Project 2025A03J4491.

References

- Josh Alman and Hantao Yu. 2025. [Fundamental limitations on subquadratic alternatives to transformers](#). In *International Conference on Learning Representations*.
- Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. 2024. [L-Eval: Instituting standardized evaluation for long context language models](#). In *Proceedings of the 62nd Annual Meeting of the Association*

for Computational Linguistics (Volume 1: Long Papers), pages 14388–14411, Bangkok, Thailand. Association for Computational Linguistics.

- Aryaman Arora, Neil Rathi, Nikil Rooshan Selvam, Róbert Csordás, Dan Jurafsky, and Christopher Potts. 2025. [Mechanistic evaluation of transformers and state space models](#). *arXiv preprint arXiv:2505.15105*.
- Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Ré. 2024a. [Zoology: Measuring and improving recall in efficient language models](#). *arXiv preprint arXiv:2312.04927*.
- Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, Dylan Zinsley, James Zou, Atri Rudra, and Christopher Ré. 2024b. [Simple linear attention language models balance the recall-throughput tradeoff](#). *arXiv preprint arXiv:2402.18668*.
- Simran Arora, Aman Timalsina, Aaryan Singhal, Benjamin Spector, Sabri Eyuboglu, Xinyi Zhao, Ashish Rao, Atri Rudra, and Christopher Ré. 2024c. [Just read twice: Closing the recall gap for recurrent language models](#). *arXiv preprint arXiv:2407.05483*.
- Sangmin Bae, Bilge Acun, Chien-Yu Lin, Haroun Habeeb, Seungyeon Kim, Liang Luo, Junjie Wang, and Carole-Jean Wu. 2025. [Hybrid architectures for language models: Systematic analysis and design insights](#). *arXiv preprint arXiv:2510.04800*. ArXiv:2510.04800v3.
- Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2023. [LongBench: A bilingual, multitask benchmark for long context understanding](#). *arXiv preprint arXiv:2308.14508*.
- Abhinaba Basu. 2026. [When does content-based routing work? representation requirements for selective attention in hybrid sequence models](#). *arXiv preprint arXiv:2603.20997*.
- 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](#). *arXiv preprint arXiv:2405.04517*.
- Ali Behrouz, Zeman Li, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. 2026. [Memory caching: RNNs with growing memory](#). *arXiv preprint arXiv:2602.24281*.
- Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. [Longformer: The long-document transformer](#). *arXiv preprint arXiv:2004.05150*.
- Aviv Bick, Tobias Katsch, Nimit Sohoni, Arjun Desai, and Albert Gu. 2025a. [Llamba: Scaling distilled recurrent models for efficient language processing](#). *arXiv preprint arXiv:2502.14458*.

- Aviv Bick, Kevin Y. Li, Eric P. Xing, J. Zico Kolter, and Albert Gu. 2024. [Transformers to SSMs: Distilling quadratic knowledge to subquadratic models](#). In *Advances in Neural Information Processing Systems*.
- Aviv Bick, Eric P. Xing, and Albert Gu. 2025b. [Understanding the skill gap in recurrent language models: The role of the Gather-and-Aggregate mechanism](#). In *International Conference on Machine Learning*.
- Aviv Bick, Eric P. Xing, and Albert Gu. 2026. [Retrieval-aware distillation for Transformer-SSM hybrids](#). *arXiv preprint arXiv:2602.11374*.
- Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, and 9 others. 2022. [Improving language models by retrieving from trillions of tokens](#). In *International Conference on Machine Learning*.
- 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, and 43 others. 2024. [RecurrentGemma: Moving past transformers for efficient open language models](#). *arXiv preprint arXiv:2404.07839*.
- Aydar Bulatov, Yuri Kuratov, and Mikhail Burtsev. 2022. [Recurrent memory transformer](#). *arXiv preprint arXiv:2207.06881*.
- Loïc Cabannes, Maximilian Beck, Gergely Szilvasy, Matthijs Douze, Maria Lomeli, Jade Copet, Pierre-Emmanuel Mazaré, Gabriel Synnaeve, and Hervé Jégou. 2025. [Short window attention enables long-term memorization](#). *arXiv preprint arXiv:2509.24552*.
- Aditya Chattopadhyay, Elvis Nunez, Prannay Kaul, Benjamin Bowman, Evan Becker, Luca Zancato, David Thomas, Wei Xia, and Stefano Soatto. 2026. [Priming: Hybrid state space models from pre-trained transformers](#). *arXiv preprint arXiv:2605.08301*.
- Yingfa Chen, Zhen Leng Thai, Zihan Zhou, Zhu Zhang, Xingyu Shen, Shuo Wang, Chaojun Xiao, Xu Han, and Zhiyuan Liu. 2026. [Hybrid linear attention done right: Efficient distillation and effective architectures for extremely long contexts](#). *arXiv preprint arXiv:2601.22156*.
- Xin Cheng, Rui Tian, Wangding Zeng, Damai Dai, Qinyu Chen, Bingxuan Wang, Zhenda Xie, Kezhao Huang, Xingkai Yu, Chengqi Deng, Shangyan Zhou, Chenggang Zhao, Zhewen Hao, Yukun Li, Han Zhang, Zhengyan Zhang, Yixu Wei, M. Y. Xu, Huishuai Zhang, and 2 others. 2026. [Conditional memory via scalable lookup: A new axis of sparsity for large language models](#). *arXiv preprint arXiv:2601.07372*.
- Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. [Generating long sequences with sparse transformers](#). *arXiv preprint arXiv:1904.10509*.
- Nicola Muca Cirone, Antonio Orvieto, Benjamin Walker, Cristopher Salvi, and Terry Lyons. 2024. [Theoretical foundations of deep selective state-space models](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. 2019. [Transformer-XL: Attentive language models beyond a fixed-length context](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988. Association for Computational Linguistics.
- Tri Dao. 2024. [FlashAttention-2: Faster attention with better parallelism and work partitioning](#). In *International Conference on Learning Representations*.
- Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. [FlashAttention: Fast and memory-efficient exact attention with IO-awareness](#). In *Advances in Neural Information Processing Systems*, volume 35.
- Tri Dao and Albert Gu. 2024. [Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality](#). In *International Conference on Machine Learning*.
- 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](#). *arXiv preprint arXiv:2402.19427*.
- Yichuan Deng, Zhao Song, Jing Xiong, and Chiwun Yang. 2024. [How sparse attention approximates exact attention? your attention is naturally \$n^c\$ sparse](#). *arXiv preprint arXiv:2404.02690*.
- Jiayu Ding, Shuming Ma, Li Dong, Xingxing Zhang, Shaohan Huang, Wenhui Wang, Nanning Zheng, and Furu Wei. 2023. [LongNet: Scaling transformers to 1,000,000,000 tokens](#). *arXiv preprint arXiv:2307.02486*.
- 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 *International Conference on Learning Representations*.
- Igor Dubinin, Antonio Orvieto, and Felix Effenberger. 2026. [Improved state mixing in higher-order and block diagonal linear recurrent networks](#). *arXiv preprint arXiv:2602.12021*.

- Parsa Ashrafi Fashi, Utkarsh Saxena, Mehdi Rezagholizadeh, Aref Jafari, Akash Haridas, Mingyu Yang, Vansh Bhatia, Guihong Li, Vikram Appia, and Emad Barsoum. 2026. [Long-context aware upcycling: A new frontier for hybrid LLM scaling](#). *arXiv preprint arXiv:2604.24715*.
- Alexander M. Fichtl, Jeremias Bohn, Josefin Kelber, Edoardo Mosca, and Georg Groh. 2025. [The end of transformers? on challenging attention and the rise of sub-quadratic architectures](#). *arXiv preprint arXiv:2510.05364*.
- Daniel Y. Fu, Tri Dao, Khaled K. Saab, Armin W. Thomas, Atri Rudra, and Christopher Ré. 2023. [Hungry hungry hippos: Towards language modeling with state space models](#). In *International Conference on Learning Representations (ICLR)*.
- Paolo Glorioso, Quentin Anthony, Yury Tokpanov, James Whittington, Jonathan Pilault, Adam Ibrahim, and Beren Millidge. 2024. [Zamba: A compact 7B SSM hybrid model](#). *arXiv preprint arXiv:2405.16712*.
- Daniel Goldstein, Eric Alcaide, Janna Lu, and Eugene Cheah. 2025. [RADLADS: Rapid attention distillation to linear attention decoders at scale](#). In *Conference on Language Modeling*.
- Riccardo Grazi, Julien Siems, Simon Schrod, Thomas Brox, and Frank Hutter. 2024. [Is mamba capable of in-context learning?](#) *arXiv preprint arXiv:2402.03170*.
- Riccardo Grazi, Julien Siems, Arber Zela, Jörg K.H. Franke, Frank Hutter, and Massimiliano Pontil. 2025. [Unlocking state-tracking in linear RNNs through negative eigenvalues](#). In *International Conference on Learning Representations*.
- Albert Gu and Tri Dao. 2023. [Mamba: Linear-time sequence modeling with selective state spaces](#). *arXiv preprint arXiv:2312.00752*.
- Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. 2020. [HiPPO: Recurrent memory with optimal polynomial projections](#). In *Advances in Neural Information Processing Systems*, volume 33.
- Albert Gu, Karan Goel, and Christopher Ré. 2022. [Efficiently modeling long sequences with structured state spaces](#). In *International Conference on Learning Representations*.
- Yuxian Gu, Qinghao Hu, Shang Yang, Haocheng Xi, Junyu Chen, Song Han, and Han Cai. 2025. [Jet-Nemotron: Efficient language model with post neural architecture search](#). In *Advances in Neural Information Processing Systems*.
- Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekish, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. [RULER: What's the real context size of your long-context language models?](#) *arXiv preprint arXiv:2404.06654*.
- Jamba Team. 2024. [Jamba-1.5: Hybrid transformer-Mamba models at scale](#). *arXiv preprint arXiv:2408.12570*.
- Samy Jelassi, David Brandfonbrener, Sham M. Kakade, and Eran Malach. 2024. [Repeat after me: Transformers are better than state space models at copying](#). In *International Conference on Machine Learning*.
- Sanjay Kariyappa and G. Edward Suh. 2026. [Side-Quest: Model-driven KV cache management for long-horizon agentic reasoning](#). *arXiv preprint arXiv:2602.22603*.
- Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. [Transformers are RNNs: Fast autoregressive transformers with linear attention](#). In *International Conference on Machine Learning (ICML)*.
- Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. [Generalization through memorization: Nearest neighbor language models](#). In *International Conference on Learning Representations*.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. [Efficient memory management for large language model serving with PagedAttention](#). In *Proceedings of the ACM Symposium on Operating Systems Principles*.
- Disen Lan, Weigao Sun, Jiaxi Hu, Jusen Du, and Yu Cheng. 2025. [Liger: Linearizing large language models to gated recurrent structures](#). *arXiv preprint arXiv:2503.01496*.
- Sangyun Lee, Sean McLeish, Tom Goldstein, and Giulia Fanti. 2026. [Do language models need sleep? offline recurrence for improved online inference](#). *arXiv preprint arXiv:2605.26099*.
- Jingdi Lei, Di Zhang, Junxian Li, Weida Wang, Kaixuan Fan, Xiang Liu, Qihan Liu, Xiaoteng Ma, Baian Chen, and Soujanya Poria. 2026. [\$\delta\$ -mem: Efficient online memory for large language models](#). *arXiv preprint arXiv:2605.12357*.
- Haoyang Li, Yiming Li, Anxin Tian, Tianhao Tang, Zhanchao Xu, Xuejia Chen, Nicole Hu, Wei Dong, Qing Li, and Lei Chen. 2025. [A survey on large language model acceleration based on KV cache management](#). *Transactions on Machine Learning Research*.
- Yixing Li, Ruobing Xie, Zhen Yang, Xingwu Sun, Shuaipeng Li, Weidong Han, Zhanhui Kang, Di Wang, and Yu Cheng. 2026. [TransMamba: A sequence-level hybrid transformer-Mamba language model](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 31823–31833.
- Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. 2024a. [SnapKV:](#)

- LLM knows what you are looking for before generation. In *Advances in Neural Information Processing Systems*.
- Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. 2024b. [EAGLE: Speculative sampling requires rethinking feature uncertainty](#). In *International Conference on Machine Learning (ICML)*.
- Opher Lieber, Barak Lenz, Hofit Bata, Gal Cohen, Jhonathan Osin, Itay Dalmedigos, Erez Safahi, Shaked Meirum, Yonatan Belinkov, Shai Shalev-Shwartz, Omri Abend, Raz Alon, Tomer Asida, Amir Bergman, Roman Glozman, Michael Gokhman, Avashalom Manevich, Nir Ratner, Noam Rozen, and 3 others. 2024. [Jamba: A hybrid Transformer-Mamba language model](#). *arXiv preprint arXiv:2403.19887*.
- Ling Team. 2025. [Every attention matters: An efficient hybrid architecture for long-context reasoning](#). *arXiv preprint arXiv:2510.19338*.
- Jiaheng Liu, Dawei Zhu, Zhiqi Bai, Yancheng He, Huanxuan Liao, Haoran Que, Zekun Wang, Chenchen Zhang, Ge Zhang, Jiebin Zhang, Yuanxing Zhang, Zhuo Chen, Hangyu Guo, Shilong Li, Ziqiang Liu, Yong Shan, Yifan Song, Jiayi Tian, Wenhao Wu, and 18 others. 2025. [A comprehensive survey on long context language modeling](#). *arXiv preprint arXiv:2503.17407*.
- Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhao Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. 2024. [KIVI: A tuning-free asymmetric 2bit quantization for KV cache](#). In *International Conference on Machine Learning*.
- Leon Lufkin, Tomas Figliolia, Beren Millidge, and Kamesh Krishnamurthy. 2026. [Hybrid associative memories](#). *arXiv preprint arXiv:2603.22325*.
- Xuezhe Ma, Chunting Zhou, Xiang Kong, Junxian He, Liangke Gui, Graham Neubig, Jonathan May, and Luke Zettlemoyer. 2023. [MEGA: Moving average equipped gated attention](#). In *International Conference on Learning Representations*.
- William Merrill, Jackson Petty, and Ashish Sabharwal. 2024. [The illusion of state in state-space models](#). In *International Conference on Machine Learning*.
- Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ram Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, Edouard Grave, Yann LeCun, and Thomas Scialom. 2023. [Augmented language models: A survey](#). *arXiv preprint arXiv:2302.07842*.
- Felix Michalak and Steven Abreu. 2025. [Some attention is all you need for retrieval](#). *arXiv preprint arXiv:2510.19861*.
- MiniCPM Team. 2026. [MiniCPM-SALA: Hybridizing sparse and linear attention for efficient long-context modeling](#). *arXiv preprint arXiv:2602.11761*.
- MiniMax. 2025. [MiniMax-01: Scaling foundation models with lightning attention](#). *arXiv preprint arXiv:2501.08313*.
- Saptarshi Mitra, Rachid Karami, Haocheng Xu, Sitao Huang, and Hyoukjun Kwon. 2025. [Characterizing state space model and hybrid language model performance with long context](#). *arXiv preprint arXiv:2507.12442*.
- Amirkeivan Mohtashami and Martin Jaggi. 2023. [Landmark attention: Random-access infinite context length for transformers](#). *arXiv preprint arXiv:2305.16300*.
- Piotr Nawrot, Robert Li, Renjie Huang, Sebastian Ruder, Kelly Marchisio, and Edoardo M. Ponti. 2025. [The sparse frontier: Sparse attention trade-offs in Transformer LLMs](#). *arXiv preprint arXiv:2504.17768*.
- Chien Van Nguyen, Huy Nguyen, Ruiyi Zhang, Hanieh Deilamsalehy, Puneet Mathur, Viet Dac Lai, Hao-liang Wang, Jayakumar Subramanian, Ryan A. Rossi, Trung Bui, Nikos Vlassis, Franck Dernoncourt, and Thien Huu Nguyen. 2025. [Lizard: An efficient linearization framework for large language models](#). *arXiv preprint arXiv:2507.09025*.
- NVIDIA. 2025. [Nemotron-H: A family of accurate and efficient hybrid Mamba-Transformer models](#). *arXiv preprint arXiv:2504.03624*.
- Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, and 7 others. 2022. [In-context learning and induction heads](#). *Transformer Circuits Thread*.
- Adnan Omerjee, Zafeirios Fountas, Haitham Bou-Ammar, and Jun Wang. 2026. [Bottlenecked transformers: Periodic KV cache consolidation for generalised reasoning](#). In *International Conference on Learning Representations*.
- 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 *International Conference on Machine Learning*.
- 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, and 13 others. 2023. [RWKV: Reinventing RNNs for the transformer era](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077, Singapore. Association for Computational Linguistics.

- Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalk, Eric Alcaide, Stella Biderman, Eugene Cheah, Teddy Ferdinan, Haowen Hou, Kranthi Kiran GV, Satyapriya Krishna, Ronald McClelland, Jr., Niklas Muennighoff, Fares Obeid, Atsushi Saito, Guangyu Song, Haoqin Tu, Ruichong Zhang, Bingchen Zhao, and 3 others. 2024. [Eagle and finch: RWKV with matrix-valued states and dynamic recurrence](#). In *First Conference on Language Modeling*.
- 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](#). *arXiv preprint arXiv:2503.14456*.
- 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 *International Conference on Machine Learning*.
- Zhen Qin, Songlin Yang, Weixuan Sun, Xuyang Shen, Dong Li, Weigao Sun, and Yiran Zhong. 2024. [HGRN2: Gated linear RNNs with state expansion](#). In *Conference on Language Modeling*.
- Haohao Qu, Liangbo Ning, Rui An, Wenqi Fan, Tyler Derr, Hui Liu, Xin Xu, and Qing Li. 2024. [A survey of Mamba](#). *arXiv preprint arXiv:2408.01129*.
- Qwen Team. 2025. [Qwen3-next: Towards ultimate training & inference efficiency](#). Technical blog.
- Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. 2020. [Compressive transformers for long-range sequence modelling](#). In *International Conference on Learning Representations*.
- Hubert Ramsauer, Bernhard Schödl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K. Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. 2021. [Hopfield networks is all you need](#). In *International Conference on Learning Representations*.
- 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 *International Conference on Learning Representations*.
- Ricardo Buitrago Ruiz and Albert Gu. 2025. [Understanding and improving length generalization in recurrent models](#). In *International Conference on Machine Learning*.
- Yash Sarrof, Yana Veitsman, and Michael Hahn. 2024. [The expressive capacity of state space models: A formal language perspective](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- 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*, pages 9355–9366. PMLR.
- Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, and Omer Levy. 2022. [SCROLLS: Standardized comparison over long language sequences](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.
- Weigao Sun, Disen Lan, Yiran Zhong, Xiaoye Qu, and Yu Cheng. 2025a. [LASP-2: Rethinking sequence parallelism for linear attention and its hybrid](#). *arXiv preprint arXiv:2502.07563*.
- 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. 2025b. [Learning to \(Learn at test time\): RNNs with expressive hidden states](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 57503–57522. PMLR.
- 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](#). *arXiv preprint arXiv:2307.08621*.
- Yutao Sun, Zhenyu Li, Yike Zhang, Tengyu Pan, Bowen Dong, Yuyi Guo, and Jianyong Wang. 2025c. [Efficient attention mechanisms for large language models: A survey](#). *arXiv preprint arXiv:2507.19595*.
- 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*.
- 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 *Advances in Neural Information Processing Systems*, volume 30.
- Roger Waleffe, Wonmin Byeon, Duncan Riach, Brandon Norrick, Vijay Korthikanti, Tri Dao, Albert Gu, Ali Hatamizadeh, Sudhakar Singh, Deepak Narayanan, Garvit Kulshreshtha, Vartika Singh, Jared Casper, Jan Kautz, Mohammad Shoeybi, and Bryan Catanzaro. 2024. [An empirical study of Mamba-based language models](#). *arXiv preprint arXiv:2406.07887*.
- Zhongwei Wan, Ziang Wu, Che Liu, Jinfa Huang, Zhihong Zhu, Peng Jin, Longyue Wang, and Li Yuan. 2024. [LOOK-M: Look-once optimization in KV cache for efficient multimodal long-context inference](#).

- In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4065–4078, Miami, Florida, USA. Association for Computational Linguistics.
- Junxiong Wang, Daniele Paliotta, Avner May, Alexander M. Rush, and Tri Dao. 2024a. [The mamba in the llama: Distilling and accelerating hybrid models](#). In *Advances in Neural Information Processing Systems*.
- Penghao Wang, Yuhao Zhou, Mengxuan Wu, Panpan Zhang, Zhangyang Wang, and Kai Wang. 2025a. [Data efficient any Transformer-to-Mamba distillation via attention bridge](#). *arXiv preprint arXiv:2510.19266*.
- Shenran Wang, Timothy Tin-Long Tse, and Jian Zhu. 2025b. [Understanding in-context learning beyond transformers: An investigation of state space and hybrid architectures](#). *arXiv preprint arXiv:2510.23006*.
- Xindi Wang, Mahsa Salmani, Parsa Omid, Xiangyu Ren, Mehdi Rezagholizadeh, and Armaghan Eshaghi. 2024b. [Beyond the limits: A survey of techniques to extend the context length in large language models](#). *arXiv preprint arXiv:2402.02244*.
- Kaiyue Wen, Xingyu Dang, and Kaifeng Lyu. 2025. [RNNs are not Transformers \(yet\): The key bottleneck on in-context retrieval](#). In *International Conference on Learning Representations*.
- Bingheng Wu, Jingze Shi, Yifan Wu, Nan Tang, and Yuyu Luo. 2025a. [TransXSSM: A hybrid Transformer-State space model with unified rotary position embedding](#). *arXiv preprint arXiv:2506.09507*.
- Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. 2025b. [Retrieval head mechanistically explains long-context factuality](#). In *International Conference on Learning Representations*.
- Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. 2022. [Memorizing transformers](#). In *International Conference on Learning Representations*.
- Chaojun Xiao, Pengl Zhang, Xu Han, Guangxuan Xiao, Yankai Lin, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. 2024a. [InfLLM: Training-free long-context extrapolation for LLMs with an efficient context memory](#). In *Advances in Neural Information Processing Systems*.
- Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. 2025. [DuoAttention: Efficient long-context LLM inference with retrieval and streaming heads](#). In *International Conference on Learning Representations*.
- Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024b. [Efficient streaming language models with attention sinks](#). In *International Conference on Learning Representations*.
- Rui Xu, Shu Yang, Yihui Wang, Yu Cai, Bo Du, and Hao Chen. 2024. [Visual Mamba: A survey and new outlooks](#). *arXiv preprint arXiv:2404.18861*.
- Songlin Yang, Jan Kautz, and Ali Hatamizadeh. 2025. [Gated delta networks: Improving mamba2 with delta rule](#). *arXiv preprint arXiv:2412.06464*.
- 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*, volume 235 of *Proceedings of Machine Learning Research*, pages 56501–56523. PMLR.
- Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. 2024. [HELMET: How to evaluate long-context language models effectively and thoroughly](#). *arXiv preprint arXiv:2410.02694*.
- Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Yuxing Wei, Lean Wang, Zhiping Xiao, Yuqing Wang, Chong Ruan, Ming Zhang, Wenfeng Liang, and Wangding Zeng. 2025. [Native sparse attention: Hardware-aligned and natively trainable sparse attention](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23078–23097, Vienna, Austria. Association for Computational Linguistics.
- Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. [Big Bird: Transformers for longer sequences](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297.
- Zhihao Zhan, Jianan Zhao, Zhaocheng Zhu, and Jian Tang. 2025. [Overcoming long-context limitations of state-space models via context-dependent sparse attention](#). In *Advances in Neural Information Processing Systems*.
- Michael Zhang, Simran Arora, Rahul Chalamala, Alan Wu, Benjamin Spector, Aaryan Singhal, Krithik Ramesh, and Christopher Ré. 2025. [LoLCATs: On low-rank linearizing of large language models](#). In *International Conference on Learning Representations*.
- Michael Zhang, Kush Bhatia, Hermann Kumbong, and Christopher Ré. 2024a. [The hedgehog & the porcupine: Expressive linear attentions with softmax mimicry](#). In *International Conference on Learning Representations*.
- Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024b. [∞Bench: Extending long context evaluation beyond 100k tokens](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuan-dong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. 2023. [H₂O: Heavy-hitter oracle for efficient generative inference of large language models](#). In *Advances in Neural Information Processing Systems*.

Tianyu Zhao and Llion Jones. 2026. [Fast-weight product key memory](#). *arXiv preprint arXiv:2601.00671*.

Jingwei Zuo, Maksim Velikanov, Ilyas Chahed, Younes Belkada, Dhia Eddine Rhayem, Guillaume Kunsch, Hakim Hacid, Hamza Yous, Brahim Farhat, Ibrahim Khadraoui, Mugariya Farooq, Giulia Campan, Ruxandra Cojocaru, Yasser Djilali, Shi Hu, Iheb Chaabane, Puneesh Khanna, Mohamed El Amine Seddik, Ngoc Dung Huynh, and 8 others. 2025. [Falcon-H1: A family of hybrid-head language models redefining efficiency and performance](#). *arXiv preprint arXiv:2507.22448*.

Appendix Roadmap

The appendix consolidates evidence, scope, and differentiation details. Appendix A records the survey protocol and coding schema; Appendix B summarizes how cited evidence is used; Appendix C explains boundaries between resources and roles; Appendix D separates capability evidence by pressure, evidence type, and caveat; Appendix E groups architectures by allocation pattern; Appendix F compares conversion and alignment targets; Appendix G records deployment budget constraints; and Appendix H differentiates this survey from related surveys. The tables record how each paper is used and which main-text claim it supports.

A Survey Protocol and Coding Schema

This appendix records the search space and coding scheme used for the targeted survey. Table 8 summarizes the search scope, expansion procedure, inclusion rule, publication status, and evidence type used throughout the survey.

Machine-readable companion. The snapshot in `data/sea_literature_coding.csv` contains one row for every work cited by this manuscript, and `data/sea_quantitative_evidence.csv` separates selected reported effects from their SEA interpretations. Field definitions, query IDs, and source notes are documented in `data/README.md`. Query-to-paper assignments map cited works retrospectively to the published seed queries because a timestamped per-paper screening log is unavailable.

B Evidence Use Table

This appendix answers a narrower evidence-use question: what role does each paper cluster play in the survey, and what caveat limits that use? It is organized by resource and role assignment and allocation granularity within the multi-axis framework, so each row states how a paper cluster is used rather than merely listing related work. Table 9 maps each evidence theme to its representative papers, survey use, and scope caveat.

C Extended Resource and Role Taxonomy

This appendix answers the taxonomy-boundary question: how do specific mechanisms fit under the hierarchy of resource budgets and functional roles introduced in Section 4? It distinguishes the

three budgets—*active latent State*, *addressable Evidence*, and *Access*—then describes the roles placed over them: direct lookup of retained evidence, compressed state evolution, online-updated associative prediction, external lookup, and budget and control policies. For each family it asks what resource and role are realized, at what allocation granularity, and what retrieval-relevant behavior is preserved or lost. The main text in Section 4 should be read first; this appendix is a reference layer for readers who want more detail on how specific models fit into the taxonomy.

C.1 Explicit Content-Addressable Retrieval Memory

The defining property of this class is that retrieval is *content-addressed*: a query vector identifies relevant past positions by similarity, independent of their sequence position. The canonical instance is Transformer softmax attention (Vaswani et al., 2017), which scores all n past keys against each query and aggregates the corresponding values. The KV cache stores all past key-value pairs; memory grows linearly with context.

Variants within this class include:

- **Sparse and local attention:** restrict the set of positions each query may attend to, reducing the number of attended positions from $O(n)$ to $O(k)$ or $O(w)$ per query. Direct content-addressable access is retained for attended positions; coverage is traded for efficiency. Representative designs include Sparse Transformers with fixed strided or block patterns (Child et al., 2019), Longformer’s mixture of local and global attention (Beltagy et al., 2020), and BigBird’s sparse, random, and global combination (Zaheer et al., 2020). All retain content-addressable access within their attended set; the allocation decision is which positions enter the attended set.
- **Sliding-window attention:** local attention with a fixed window; used in Griffin and RecurrentGemma alongside global recurrence.
- **Shared attention modules:** a single attention module applied at multiple depths (Zamba); centralized retrieval with reduced per-layer KV cost.

C.2 Compressed-State Recurrent Memory

This class replaces the per-token KV store with a fixed-size state $h \in \mathbb{R}^d$ that is updated at each

Protocol field	Specification
Search scope	ACL Anthology, arXiv, OpenReview, Semantic Scholar, and major conference and systems proceedings, focused on 2017–2026 language-model memory work.
Seed queries	Hybrid language model; SSM and Mamba; linear and recurrent language model; long context; memory compression; retrieval head; KV cache; external memory; Transformer-to-SSM or Transformer-to-Mamba distillation.
Expansion	Snowballing from related surveys, benchmark papers, and system papers that clarify retained evidence, compressed state, online update, external lookup, or cache and access policies.
Inclusion rule	Include work informing at least one SEA dimension for autoregressive text modeling: resource role, mechanism, allocation granularity, intervention phase, memory function, or mismatch risk.
Exclusion rule	Exclude mainly vision, genomics, time-series, graph, or broad multimodal work unless it defines a boundary case for language-model memory allocation.
Provenance limitation	The historical search cutoff and candidate-level exclusion decisions were not retained. Coverage is targeted and should not be interpreted as exhaustive or PRISMA-grade reproducible.
Coding schema	Resource role; mechanism; allocation granularity; intervention phase; memory function; mismatch risk; benchmark or probe type; publication status; evidence type.
Publication status	Peer-reviewed paper; preprint; technical report or blog.
Evidence type	Theory; benchmark; mechanistic analysis. Strong claims in the main text draw on peer-reviewed, formal, benchmark, or convergent evidence; a single recent report is treated as preliminary.

Table 8: Survey protocol and coding schema.

step. Arbitrary-token lookup over the full prefix is not generally supported; the guarantee is efficient inference with $O(1)$ per-step state memory.

Historical lineage:

- Transformer-XL (Dai et al., 2019): introduced segment-level recurrence by caching and reusing per-token hidden states from previous segments; the memory is a hidden-state sequence rather than a compressed segment summary or literal KV cache.
- HiPPO (Gu et al., 2020): optimal polynomial projection for compressing history into recurrent coefficients.
- S4 (Gu et al., 2022): structured parameterization of the state transition via diagonal-plus-low-rank matrices.
- Mamba (Gu and Dao, 2023): selective state transitions that gate how much each input token is absorbed into state.
- Mamba-2 and SSD (Dao and Gu, 2024): reformulation as structured matrix products; connects to linear attention.
- RWKV (Peng et al., 2023) and RetNet (Sun et al., 2023): alternative parameterizations using time-mixing and retention.
- Eagle and Finch (Peng et al., 2024): RWKV variants with matrix-valued states and dynamic recurrence.

- xLSTM (Beck et al., 2024): matrix-valued cell state extending LSTM to higher-dimensional compressed memory.
- StateMix (Dubinin et al., 2026): higher-order and block-diagonal mixing within a fixed-size recurrent state; it does not add external or growing memory.

A recurring challenge for this class is length generalization: models trained on shorter sequences often fail to maintain coherent recurrent state at longer test-time horizons. Ruiz and Gu (2025) analyze the sources of this failure and propose strategies that improve length generalization in recurrent models while preserving their $O(1)$ -memory streaming advantage.

C.3 Online-Updated Active-State Memory

This class is a structured active-state role: it generalizes compressed-state models by allowing the recurrent state to be selectively *rewritten* rather than only accumulated. The state is typically a matrix-valued accumulator $S \in \mathbb{R}^{d \times d}$, and updates are governed by a learned rule (delta rule, gradient step, or gated difference).

Historical lineage:

- Linear attention (Katharopoulos et al., 2020): replaces softmax attention with a feature-map kernel whose associative matrix accumulation admits a recurrent implementation. Schlag

et al. (2021) formalize this as *fast weight programming*: linear layers are “programmed” by an outer-product write rule, making the connection to associative memory explicit.

- Induction heads (Olsson et al., 2022): mechanistic analysis of attention heads that implement “copy one token after a matching prefix”; illustrates that specific attention heads carry retrieval function, motivating the mismatch problem in Section 6.
- H3 (Fu et al., 2023): combines a shift SSM (for local dependencies) with a diagonal SSM (for long-range dependencies) plus two retained attention layers; precursor to modern hybrid designs at the layer level.
- Gated Linear Attention (GLA) (Yang et al., 2024): adds data-dependent gating for structured forgetting.
- Based (Arora et al., 2024b): simplified linear attention with an explicit tradeoff between recall and efficiency.
- DeltaNet and Gated DeltaNet (Yang et al., 2025): delta-rule updates; selectively rewrite associations in the matrix state.
- Test-Time Training (TTT) (Sun et al., 2025b): treats the recurrent state as the parameters of a small online-trained model; K-V pairs are “memorized” via gradient updates.
- Fast-weight product-key memory (Zhao and Jones, 2026): sparse episodic fast-weight updates; bridges online-updated recurrent memory and external memory.

The key distinction from simpler compressed-state models is that online-updated models can *selectively retain* specific associations, making them more retrieval-relevant—but they remain capacity-limited by the same active-state budget.

C.4 External and Persistent Memory

External memory externalizes addressable Evidence beyond model parameters and activation state, shifting the primary constraint from active-state capacity to storage and to the Access budget for retrieval and integration. The retrieval cost shifts from compute-at-attention-time to I/O latency and index infrastructure. External retrieval systems exemplify this pattern for language models.

Hybrid Associative Memories (Lufkin et al., 2026) explore a selective externalization strategy: hard-to-predict tokens are added to a KV cache, while the rest is handled by a recurrent compressive base. Engram (Cheng et al., 2026) is a boundary case for this survey: it adds conditional static lookup memory for language models, showing that the SEA framework extends beyond SSM–attention hybrids without changing the survey’s central taxonomy.

C.5 Boundary Cases and Design Tradeoffs

Several models occupy positions at the boundary between mechanism classes:

- **DuoAttention** (Xiao et al., 2025): partitions attention heads into retrieval heads (full KV) and streaming heads (compressed KV); a head-aware cache-allocation boundary case *within* the attention class.
- **RWKV-7** (Peng et al., 2025) and **Eagle and Finch** (Peng et al., 2024): expressive dynamic state evolution and matrix-valued recurrent states; approach online-updated recurrent memory behavior while maintaining efficient recurrent inference.
- **Bottlenecked Transformers** (Omerjee et al., 2026): periodic KV abstraction; attention with periodic compression; boundary between full attention and compressed-state.
- **Fundamental limits** (Alman and Yu, 2025): formal lower bounds on the capabilities of subquadratic alternatives to Transformers under specified computational models; motivates caution when replacing attention-like access with cheaper sequence mechanisms.
- **Formal-language expressivity of SSMs** (Sarraf et al., 2024): studies SSM expressivity relative to Transformers and RNNs. The SEA mapping from these limits to memory roles and retrieval is our inference.
- **Theoretical expressiveness of SSMs** (Cirone et al., 2024): analyzes selective SSM hidden states and expressivity. The SEA mapping from these results to compressed-State retrieval roles is our inference.
- **Modern Hopfield Networks** (Ramsauer et al., 2021): reformulates softmax attention as a Hopfield energy update, explicitly connecting

attention to associative memory retrieval; the memory capacity scales exponentially with the state dimension, connecting K-V lookup over retained Evidence with associative active-state roles.

- **LIGER** (Lan et al., 2025): post-training linearization of a large language model into a gated recurrent structure; provides evidence that replacement choices are layer-sensitive rather than uniformly safe.

D Capability and Benchmark Evidence Map

This appendix answers the capability-evidence question: which pressure does each benchmark or formal result support, what type of evidence is it, and what caveat limits the claim? For each capability claim in the main text, the tables separate supporting paper(s), evidence type, scope of the claim, and caveats. Recent preprints are used as representative evidence, not as sole support for strong quantitative claims.

D.1 Context-Access Capability Evidence

This subsection collects evidence for the most attention-sensitive capability pressure: direct access to specific past tokens, facts, or patterns. The rows combine formal copying and retrieval limits (Jelassi et al., 2024; Wen et al., 2025), synthetic recall tests (Arora et al., 2024a,c), mechanistic evidence for retrieval heads (Olsson et al., 2022; Bick et al., 2025b), and long-context factual lookup stress tests (Mitra et al., 2025). We use these results to support qualitative claims about resources and roles rather than single-number rankings across model families. Table 10 lists the specific evidence items and caveats behind this context-access claim.

D.2 State-Evolution Capability Evidence

State-evolution evidence targets a different pressure: maintaining and updating compressed latent state across long sequences. The relevant failures are not necessarily missed facts, but unstable transition rules, entity-state errors, or limits of particular recurrent parameterizations. We therefore separate formal state-tracking results (Merrill et al., 2024) from empirical and theoretical evidence about transition structure, such as negative eigenvalue dynamics (Grazzi et al., 2025). Table 11 summarizes the state-evolution evidence used to distinguish update-rule limits from retrieval failure.

D.3 Benchmark Protocol Notes

Benchmark rows are included because apparent architecture gaps can be amplified or hidden by protocol choices. Synthetic recall probes such as MQAR and needle-style tasks isolate addressable-evidence access, whereas LongBench, SCROLLS, RULER, and InfiniteBench-style suites mix retrieval, reasoning, summarization, and length extrapolation (Arora et al., 2024a; Bai et al., 2023; Shaham et al., 2022; Hsieh et al., 2024; Zhang et al., 2024b). The protocol notes identify which results support SEA-specific inferences and which serve as broad end-to-end stress tests. Table 12 records these benchmark roles and caveats.

Takeaway. Together these rows separate context access, state evolution, and benchmark protocol effects. The evidence supports the framework by showing that failures are not a single “long-context” deficit: they depend on whether the task stresses addressable Evidence, active-State updates, or evaluation protocol design.

E Hybrid Architecture Catalogue

This appendix answers the architecture-comparison question: which allocation pattern does each hybrid system instantiate, and what mismatch risk follows from that pattern? Rows group systems by granularity, resource and role assignment, and mismatch risk rather than by model family alone. Table 13 gives the corresponding architecture catalogue.

Takeaway. The catalogue preserves architecture breadth while making the allocation pattern the comparison unit. The main value of the table is the mismatch-risk column: similar resource savings can fail for different reasons depending on whether allocation happens at block, layer, head, schedule, query-selection, or cache-policy granularity.

F Distillation, Conversion, and Alignment Catalogue

This appendix answers the alignment question: when a Transformer is converted or distilled into a recurrent, linear-attention, or hybrid student, which behavior is actually preserved? It compares output-aligned conversion, hidden-state and attention alignment, retrieval-aware alignment, and scalable conversion through the lens of preservation of resources and their roles, especially direct access to retained evidence via G&A-type retrieval behavior. For each method the table records the align-

ment target (logit KD, hidden-state, attention-map, retrieval-behavior, or memory-dynamics), what is concretely preserved, and whether the method addresses the context-access gap or only the output gap. The rows below emphasize methods that directly affect the main-text argument. Table 14 summarizes these conversion and alignment methods.

Takeaway. The table distinguishes scalable conversion from function-preserving alignment. Output or hidden-state matching can transfer broad capability, but the framework predicts that retrieval-aware targets are needed when the role to preserve is direct context access to Evidence.

Design lesson. The trend across these methods is from *output-aligned conversion* (logit KD and hidden-state alignment: MOHAWK, Llama, Mamba in Llama), through architecture-priming conversion (Priming), toward *function-aligned conversion* (retrieval behavior: RAD; hybrid linear attention (Chen et al., 2026)). The evidence that logit alignment alone may not preserve context-access capabilities motivates retrieval-aware objectives as the next step in distillation design.

G Systems and Deployment Evidence

This appendix answers the systems question: which deployment constraint sets the effective budget for memory allocation? Each entry is framed around the *budget constraint* it imposes or relaxes, treating deployment requirements as the source of the allocation budget that drives architectural choice. For each method the table records: the memory bottleneck targeted (KV-cache capacity, KV-cache bandwidth, recurrent-state memory, distributed-training communication, or system-level routing cost); the budget mechanism (what is evicted, compressed, or offloaded); and whether the design interacts with the main-text allocation granularities (head-level, layer-level, system-level, or cache-policy). Methods that reduce generic FLOPs without targeting a specific memory budget are noted but deprioritized; the focus is on methods that establish a concrete cost model for hybrid architecture design. Table 15 records the deployment evidence and its caveats.

Takeaway. Systems methods define the budgets within which architecture choices operate. The table separates methods that make attention cheaper from methods that change the State–Evidence split, preventing kernel or cache advances from being mistaken for architectural allocation.

Key design lesson. Kernel-level and cache-management methods are *complementary* to architectural hybridization: FlashAttention improves the per-step cost of attention but does not remove KV-cache growth; hybrid architectures remove most KV-cache growth but benefit additionally from FlashAttention within their attention layers. The two strategies should be combined, not treated as alternatives.

H Related Surveys and Differentiation

This appendix answers the differentiation question: what does this survey add beyond existing SSM and Mamba, long-context, efficient-attention, and augmented-LM surveys? The organizing claim: existing surveys primarily catalogue architectures, applications, or efficiency techniques; this survey provides a memory-allocation *framework* for reasoning about *why* hybridization arises and *how* design choices allocate active State, addressable Evidence, and the Access budget to retrieval, updating, and policy-control roles. Table 16 summarizes the comparison against neighboring survey lines.

Takeaway. The comparison shows that prior surveys cover important neighboring literatures, but usually organize them by architecture family, efficiency method, or context-extension technique. The present survey’s added value is to link resource budgets to functional roles and thereby clarify why hybridization arises.

Novelty statement. This survey contributes a *function-preserving memory-allocation framework* combining capability-pressure analysis (Section 3), resource budgets mapped to functional roles (Section 4), and a multi-axis design map (Section 5). Together, these elements explain hybridization as a resource-allocation response and distinguish this survey from neighboring work.

Evidence theme	Representative papers	Use in survey	Scope and caveat
Mechanistic retrieval gap	G&A and retrieval-head analyses (Bick et al., 2025b; Wu et al., 2025b; Michalak and Abreu, 2025); mechanistic evaluations (Arora et al., 2025; Wang et al., 2025b)	Supports Sections 3 and 6: context access depends on retrieval-specialized attention components, so coarse hybridization can preserve perplexity while losing retrieval function.	Recent mechanistic evidence; use strongest claims for qualitative mismatch and head-function arguments rather than exact universal laws.
Theoretical unity and capability separation	SSD and its structured-matrix view (Dao and Gu, 2024); RNN-Transformer separation (Wen et al., 2025); copying and subquadratic limits (Jelassi et al., 2024; Alman and Yu, 2025)	Grounds Section 4: attention and SSMS can share algebraic structure while differing in addressable Evidence, latent-State capacity, and the Access budget.	Formal models differ from deployed LLMs; claims are used to motivate capability pressure, not to declare blanket impossibility.
Compressed-state sequence modeling	S4, HiPPO, and Mamba line (Gu et al., 2022; Gu and Dao, 2023; Gu et al., 2020); recurrent baselines and bounded-state mixing (Peng et al., 2023; Sun et al., 2023; Beck et al., 2024; Qin et al., 2024; Peng et al., 2025; Dubinin et al., 2026)	Defines compressed relational or dynamical state as the active latent-state budget's main functional role and supplies the efficiency anchor for hybrid architectures.	Strong for streaming and compression; direct content-addressable access remains task-dependent and often requires additional evidence.
Online-updated active-state role	Linear attention and gated updates (Katharopoulos et al., 2020; Yang et al., 2024, 2025); TTT and fast-weight variants (Sun et al., 2025b; Zhang et al., 2024a; Zhao and Jones, 2026)	Supports the online-updated predictor and associative-map lens as a structured active-state role in Section 4 and the K-V allocation open problem.	These methods store learned statistics or predictor parameters, not a full token-level KV table; retrieval fidelity depends on update capacity.
Emerging State construction	δ -mem (Lei et al., 2026); sleep-like consolidation (Lee et al., 2026)	Distinguishes an added compact associative State pathway from extra pre-eviction State-construction compute under a fixed representation.	Both are 2026 preprints in specific model and task settings; neither result restores arbitrary evicted token-level K-V Evidence.
Capability-pressure evidence	MQAR and recall benchmarks (Arora et al., 2024a); Mamba ICL studies (Grazzi et al., 2024; Park et al., 2024); state-tracking work (Merrill et al., 2024; Grazi et al., 2025; Ruiz and Gu, 2025)	Supports Section 3: context access, smoother ICL regimes, and state evolution require different resources and roles.	Evidence is heterogeneous across theory, synthetic tasks, and empirical benchmarks; avoid hard checkmark-style claims.
Layer- and block-level hybrid allocation	Interleaving and production hybrids (Lieber et al., 2024; Jamba Team, 2024; Ren et al., 2025; NVIDIA, 2025; Gu et al., 2025; MiniMax, 2025; Qwen Team, 2025; MiniCPM Team, 2026); long-context upcycling (Fashi et al., 2026); local-global hybrids (De et al., 2024; Botev et al., 2024)	Supports the Section 5 claim that current systems mostly allocate direct access to retained Evidence at layer or block granularity.	Attention ratio and placement are not equivalent to retrieval-function preservation; detailed per-system comparison is in Appendix E.
Head-level, shared, scheduled, and query-conditioned allocation	Head-level designs (Dong et al., 2025; Zuo et al., 2025); shared or amortized attention (Glorioso et al., 2024); position- and layer-scheduled switching (Li et al., 2026); static periodic hybrid integration (Wu et al., 2025a); query-conditioned sparse evidence selection (Zhan et al., 2025); allocation analysis (Ling Team, 2025; Waleffe et al., 2024)	Shows that allocation can occur within layers, through reusable attention, at scheduled token and layer boundaries, or through query-conditioned selection of retained Evidence.	Static schedules and query-conditioned evidence selection do not directly supervise future retrieval relevance or implement content-dependent attention-recurrence module routing; mismatch risk remains central.
Training-time alignment and conversion	Conversion and distillation methods (Bick et al., 2024, 2026; Goldstein et al., 2025; Bick et al., 2025a; Wang et al., 2024a; Zhang et al., 2025; Nguyen et al., 2025; Chen et al., 2026; Wang et al., 2025a; Lan et al., 2025; Chattopadhyay et al., 2026)	Supports Section 6: replacing attention requires deciding which teacher behavior, heads, states, or retrieval traces to preserve.	Output-level alignment may not preserve context-access behavior; retrieval-aware objectives are promising but recent.
Systems and cache-budget evidence	Exact-attention kernels (Dao et al., 2022; Dao, 2024); distributed linear-attention training (Sun et al., 2025a); sparse-attention and cache policies (Xiao et al., 2024b,a, 2025; Wan et al., 2024; Deng et al., 2024; Yuan et al., 2025; Nawrot et al., 2025; Cabannes et al., 2025); comparative long-context systems characterization (Mitra et al., 2025); speculative decoding (Li et al., 2024b)	Supports Section 5 by separating physical IO, distributed-training, inference characterization, decoding, and evidence-retention interventions that impose different resource budgets on hybrid systems.	Only the sparse-attention and cache-policy subset directly changes which retained Evidence is addressable; kernel, training-parallelism, characterization, and speculative-decoding evidence must not be read as cache allocation.
External and emerging memory regimes	External and hybrid associative memory candidates (Zhao and Jones, 2026; Lufkin et al., 2026; Behrouz et al., 2026)	Supports open problems around sparse retained Evidence, growing recurrent memory, and hybrid streaming associative memory.	Recent or speculative evidence; treated as future design space rather than solved K-V allocation.

Table 9: Evidence roles and caveats used in the survey.

Evidence item	Source and type	Claim supported	Caveat
Exact copying	Jelassi et al. (2024) Theory and empirical	Fixed-state models cannot copy from arbitrary positions as $n \rightarrow \infty$.	Formal model assumes unbounded n ; practical impact varies with context length.
Associative recall (MQAR)	Arora et al. (2024a) Empirical benchmark	SSMs and linear attention underperform attention on MQAR across model sizes.	Performance gap narrows with larger state; avoid relying on exact numeric deltas.
In-context retrieval (general)	Wen et al. (2025) Theory and empirical	Sublinear-memory RNNs cannot perfectly solve several retrieval-dependent tasks under the paper’s formal assumptions.	Scope includes Mamba, RWKV, and linear Transformers; this is not a blanket claim about deployed models.
ICL with smooth targets	Grazzi et al. (2024) Empirical	Mamba performs ICL for low-dimensional smooth function classes.	Does not imply discrete content-addressable lookup or arbitrary label mapping.
Retrieval-heavy ICL	Bick et al. (2025b) Mechanistic and empirical	G&A heads appear critical for retrieval-heavy ICL and are absent in pure SSMs.	Boundary cases such as TTT or DeltaNet require caution.
Long-context factual lookup	Mitra et al. (2025) Long-context benchmarks	SSM and hybrid models show factual-retrieval drops at very long contexts.	Exact deltas are benchmark-dependent; needle-style tasks may favor attention.
Recall gap closure (JRT-RNN)	Arora et al. (2024c) Empirical	Just Read Twice substantially narrows recurrent-model recall gaps by adding a second sequential pass.	Holds for specific task formats and evaluated lengths; it does not establish arbitrary lookup under all query orders or contexts.
Induction head mechanism	Olsson et al. (2022) Mechanistic analysis	Induction heads are dedicated attention circuits for in-context pattern copying.	Findings are for Transformer attention; analogous hybrid circuits remain an open question.

Table 10: Context-access capability evidence.

Evidence item	Source and type	Claim supported	Caveat
Entity tracking	Merrill et al. (2024) Formal-language theory	Common SSM parameterizations can share bounded-depth Transformer limits for state tracking.	Idealized computational model; not a blanket claim about arbitrary gated RNNs.
Algorithmic state transitions	Merrill et al. (2024) Theoretical	Common SSMs or constrained state-space updates may fail on compositional transition sequences.	Scope is update-rule constraints, not arbitrary nonlinear recurrent computation.
Negative-eigenvalue state tracking	Grazzi et al. (2025) Empirical and theoretical	Negative eigenvalues in linear RNNs expand state-tracking capability.	State tracking depends on transition structure, not recurrence alone.

Table 11: State-evolution capability evidence.

Benchmark	Paper	What it measures	Caveat
MQAR (Multi-Query Assoc. Recall)	Arora et al. (2024a)	Associative recall fidelity across context lengths	Isolates discrete recall; does not measure broader language quality
Recall benchmark (Zoology suite)	Arora et al. (2024a)	Recall precision for SSM vs. attention families	Useful for mechanism stress tests; exact deltas are setup-dependent
LongBench	Bai et al. (2023)	Multi-task long-context comprehension (bilingual)	End-to-end benchmark; task mixture can obscure specific retrieval failures
LRA	Tay et al. (2021)	Efficient-sequence-model stress tests over long inputs	Useful historical benchmark; less aligned with modern generative LMs
SCROLLS	Shaham et al. (2022)	Long-document summarization, QA, and NLI-style tasks	Task realism improves, but memory function is less isolated
Needle-in-a-haystack	Various; Mitra et al. (2025)	Long-context factual retrieval (synthetic)	Standard protocol; numbers should be treated cautiously
RULER	Hsieh et al. (2024)	Parametric evaluation of “real” context window size under retrieval pressure	Controlled benchmark; hybrid vs. pure-attention gaps measurable
L-Eval, InfiniteBench, and HELMET	An et al. (2024); Zhang et al. (2024b); Yen et al. (2024)	Long-context application suites and 100K+ context stress tests	Broader coverage; still needs language, domain, and function tagging

Table 12: Benchmark protocol notes.

Allocation pattern	Systems	Resource and role assignment	Mismatch risk and caveat
Block-local retained-evidence access plus global recurrence	Griffin (De et al., 2024); RecurrentGemma (Botev et al., 2024)	Local-window KV provides direct access to recent K-V evidence while gated recurrence compresses the global tail.	Attractive for streaming and state evolution, but far-past retrieval depends on compressed State quality.
Periodic layer interleaving	Jamba, Jamba 1.5 (Lieber et al., 2024; Jamba Team, 2024); Samba (Ren et al., 2025); Nemotron-H, Jet-Nemotron (NVIDIA, 2025; Gu et al., 2025); MiniMax-01 (MiniMax, 2025); Qwen3-Next (Qwen Team, 2025); MiniCPM-SALA (MiniCPM Team, 2026)	Full-attention, sparse-attention, gated-attention, or linear-attention layers act as retrieval anchors inside mostly recurrent or efficient-mixer stacks.	Attention ratio is not enough: task-agnostic placement can miss retrieval-critical components (Bick et al., 2025b; Ling Team, 2025); Qwen3-Next should be read as technical-blog evidence.
Shared or amortized attention	Zamba (Glorioso et al., 2024)	Reusable or amortized attention mechanisms reduce per-layer attention cost while preserving some direct retained-evidence access.	Sharing or amortization may under-cover layer- or token-specific retrieval, so coverage should be evaluated under harder recall tasks.
Head-level fusion	Hymba (Dong et al., 2025); Falcon-H1 (Zuo et al., 2025)	Softmax heads retain K-V evidence while recurrent or SSM heads carry compressed dynamics within a layer.	The split helps only if attention heads align with retrieval-critical functions; current designs rarely supervise this alignment directly.
Position-scheduled switching	TransMamba (Li et al., 2026)	Predefined layer-specific transition points switch token positions from attention to SSM computation.	The schedule varies by layer and position, but is not content-dependent and does not predict future relevance.
Static periodic integration	TransXSMM (Wu et al., 2025a)	A fixed 7:1 SSM-to-attention sub-layer pattern inserts attention into an SSM-dominant stack.	This is architectural placement rather than an adaptive router.
Query-conditioned sparse evidence	HAX and CDSA (Zhan et al., 2025)	Query-key context guides sparse selection of retained Evidence inside a fixed SSM-plus-sparse-attention block.	Selection is content-dependent, but it neither routes between modules nor predicts future retrieval relevance.
Long-context upcycling	HyLo (Fashi et al., 2026); MiniCPM-SALA (MiniCPM Team, 2026)	Converts or continues pretrained Transformer models into hybrids with sparse or efficient attention and linear or SSM-style blocks.	Recent preprint evidence; useful for intervention-phase analysis but still needs function-specific tests.
Fine-grained growing memory	Memory Caching (Behrouz et al., 2026); hybrid associative memory (Lufkin et al., 2026); fast-weight product-key memory (Zhao and Jones, 2026)	Cached recurrent checkpoints, selective KV growth, or sparse fast-weight access approach fine-grained memory allocation.	These approaches inform K-V allocation; adaptive Transformer K-V-pair selection remains open.
Function-aware analysis and baselines	Every Attention Matters (Ling Team, 2025); HGRN2 (Qin et al., 2024); empirical Mamba and hybrid studies (Waleffe et al., 2024)	Analysis and recurrent baselines clarify which layers or mechanisms matter when attention is reduced.	Supports function-specific analysis of allocation policies.

Table 13: Hybrid architecture catalogue by allocation pattern.

Method	Year	Alignment target	Preserved behavior	Scale and status	Caveat
MOHAWK (Bick et al., 2024)	NeurIPS 2024	Logit KD; weight init.; SSD approx.	General capability; retrieval evidence remains preliminary	1.3B teacher; 1.5B student	Retrieval gap not directly measured
RAD (Bick et al., 2026)	2026 preprint	Retrieval-behavior alignment	G&A-type context-access behavior	Recent preprint	Retrieval target explicit; full evaluation preliminary
Goldstein et al. (2025)	COLM 2025	Low-token-cost fine-tuning	General capability at scale	Large-scale conversion	Retrieval-specific coverage remains limited
Priming (Chattopadhyay et al., 2026)	2026 preprint	Layer selection; weight transfer; layerwise alignment	Hybrid SSM conversion and SSM-layer comparison	8B–32B; 128K context	Recent preprint; function-specific preservation remains indirect
Llamba (Bick et al., 2025a)	2025 preprint	MOHAWK-style Transformer-to-Mamba distillation	General capability and efficient recurrent deployment	1B–8B	Does not explicitly preserve attention heads or G&A behavior
AttBridge (Wang et al., 2025a)	2025 preprint	Data-efficient Transformer-to-Mamba conversion	General capability; small-data conversion efficiency	Small-to-medium scale	Context-access preservation not directly targeted
Mamba in the Llama (Wang et al., 2024a)	NeurIPS 2024	SeqKD (pseudo-labels), word-level KL, and DPO	General capability	Up to 8B	No retrieval-specific alignment
LoLCATs (Zhang et al., 2025)	ICLR 2025	Attention transfer to sliding-window + linear attention; LoRA	General capability; low-cost conversion	Up to 405B	Retrieval role may be diluted by linearization
Lizard (Nguyen et al., 2025)	2025	Efficient linearization framework	Reduced calibration cost	Recent framework	Retrieval-specific evidence limited

Table 14: Distillation, conversion, and alignment catalogue.

Method	Bottleneck	Mechanism	Gain	Source	Caveat
FlashAttention (Dao et al., 2022)	Attention memory bandwidth	IO-aware kernel fusion; avoids materializing the attention matrix	Kernel-level	NeurIPS 2022	Kernel-only; preserves dense attention
FlashAttention-2 (Dao, 2024)	Attention memory bandwidth	Improved parallelism and tiling	Kernel-level	ICLR 2024	Kernel-only; does not change evidence allocation
LASP-2 (Sun et al., 2025a)	Distributed training memory for linear attention	Sequence parallelism with hybrid-aware communication	Distributed-training system	2025 preprint	Training-system support, not an allocation rule
StreamingLLM (Xiao et al., 2024b)	KV-cache for streaming	Attention sink and sliding window; no KV cache growth	Cache-management	2023	Cache policy; does not alter model architecture
InfLLM (Xiao et al., 2024a)	KV-cache at long context	Block-level KV memory management (training-free)	Cache-management	2024	Retrieval policy outside core architecture
DuoAttention (Xiao et al., 2025)	KV-cache (retrieval vs. streaming heads)	Full KV only for retrieval heads; compressed KV for streaming	Architectural and cache	ICLR 2025	Depends on correct head classification
Look-M (Wan et al., 2024)	Multimodal KV cache	Key merging for visual tokens	Cache-management	2024	Multimodal-boundary evidence
Sparse attention (n^C theory) (Deng et al., 2024)	Attended-token budget	Natural n^C sparsity reduces attended tokens	Architectural (theory)	2024	Theoretical sparsity model; use qualitatively
EAGLE (Li et al., 2024b)	Autoregressive decoding latency	Speculative decoding via a lightweight draft model; accepts or rejects proposed tokens using the target model’s KV cache	Latency $2.7\text{--}3.5\times$; throughput $2\times$ (LLaMA2-Chat 70B)	ICML 2024	Decoding-only speedup; not memory allocation
Hybrid architectures	KV-cache at long context	Shift majority of computation to $O(1)$ recurrent state	Architectural	Jamba, Griffin, Nemotron-H	Architecture-level budget tradeoff

Table 15: Systems and deployment evidence.

Survey	Year	Organizing principle	This survey's difference
"A Survey of Mamba" Qu et al. (2024)	2024	SSM and Mamba architecture evolution and applications	Resource budgets and allocation patterns across hybrid families form our comparison axes
"A Comprehensive Survey on Long Context Language Modeling" Liu et al. (2025)	2025	Long-context techniques covering attention, compression, and retrieval	We organize these techniques by gaps between available resources and required functions
"Visual Mamba: A Survey and New Outlooks" Xu et al. (2024)	2024	Mamba applications in computer vision	We focus on language-model retrieval and use vision hybrids only as boundary cases
"Efficient Attention Mechanisms for Large Language Models: A Survey" Sun et al. (2025c)	2025	Linear and sparse attention mechanisms, including hardware-aware efficient attention	Efficient attention is one memory-allocation substrate within the broader SEA taxonomy
"Augmented Language Models: a Survey" Mialon et al. (2023)	2023	External tools, reasoning modules, and non-parametric augmentation	We include externalized evidence as an addressable-evidence regime, while keeping architecture-level memory allocation central
"A Survey on Large Language Model Acceleration based on KV Cache Management" Li et al. (2025)	2025	Token-, model-, and system-level KV-cache management	We use cache management as the system-policy end of the same allocation taxonomy
"Beyond the Limits" Wang et al. (2024b)	2024	Techniques for extending LLM context length	We connect long-context methods to required resources and functional roles
"The End of Transformers? On Challenging Attention and the Rise of Sub-Quadratic Architectures" (Fichtl et al., 2025)	2025	Survey of sub-quadratic attention, RNNs, SSMs, and hybrids	Our framework compares behavioral requirements and capability pressures

Table 16: Related surveys and differentiation.