# The Evolution of Attention in Large Language Models

## Mechanisms, Trade-offs, and Emerging Trends

Paper by Zhentao Tan, Jingyi Shen, Yanbo Li,
Yao Liu, Yue Wu, and Jieping Ye

Alibaba Token Hub, Alibaba Group

Independent reading presentation | October 4, 2026
arXiv:2609.39661v1

The design unit is the lifecycle of contextual memory.

# Attention efficiency involves three different costs

**Prefill**

Dense causal interactions:

$$O(L^2 d)$$

Many query–key pairs.

**Decoding**

Per-step attention work:

$$O(Ld)$$

Growing KV traffic.

**Retrieval quality**

Longer context expands the candidate set.

Relevant evidence can be harder to recover.

**Survey question:** how do architectures organize memory to balance these costs and capabilities?

Paper, Secs. 1 and 4, PDF pp. 4 and 15. $L$: sequence length; $d$: attention width.

# Five functions make architectural changes comparable

| Function | Question |
|---|---|
| Representation | What survives, and in what form? |
| Update | What is appended, retained, corrected, or merged? |
| Access | Which stored units may participate in this read? |
| Readout | How does the query extract their information? |
| Integration | How do completed readouts form the output? |

These are analytical roles. One operation may change several roles, and the implementation order can vary.

Paper, Sec. 2, Eqs. (1)–(5), Fig. 1, PDF pp. 6–10.

# History coverage and token addressability are different

## Explicit memory

$$\{(k_i, v_i)\}_{i \le t}$$

The query can select individual entries.

Head/channel sharing reduces the cost per token.

Summaries and slots may lose original token identity.

Paper, Sec. 2.1, PDF p. 7; Secs. 3.2 and 5.

## Recurrent state

$$S_t = f(S_{t-1}, x_t)$$

The entire history may influence a bounded state.

Original tokens are usually no longer separately selectable.

Capacity and interference govern recovery.

# KV compression targets heads, channels, or depth

$$M_{\mathrm{KV}} = 2BTL_{\mathrm{KV}}H_{\mathrm{KV}}d_h s$$

- **MQA/GQA:** reduce stored KV heads $H_{\mathrm{KV}}$.
- **MLA:** cache a compact latent plus a positional component.
- **CLA:** reduce independent KV-producing layers $L_{\mathrm{KV}}$.

## Illustrative KV-only calculation

At 32K tokens, 32 layers, $d_h = 128$, 2-byte scalars, $B = 1$:

32 KV heads: **16 GiB** $\rightarrow$ 8 KV heads: **4 GiB**
With 4-layer KV sharing: **1 GiB**.

Paper, Sec. 3.1, Eq. (7), PDF pp. 11–12. Example calculated for this presentation; $T = 32768$. All three retain linear growth in $T$ when all tokens are cached.

# Memory compression and output gating change different things

- **Fixed-capacity memory:** RMT/FAM latent states or Trellis/Lattice KV slots overwrite or consolidate history.
- **Dynamic resolution:** recent tokens stay precise; older context becomes summaries or compressed entries.
- **Readout modulation:** change scores or attention maps, as in Differential Attention and FoX.
- **Integration modulation:** gate or route completed head outputs, as in Gated Attention and MoH.

A post-readout gate cannot remove KV loads and score computations that already occurred.

Paper, Secs. 3.2–3.3, PDF pp. 12–14. Actual savings depend on operations skipped by the implementation.

# Sparse attention budgets access to represented memory

$$\mathcal{I}_t = \mathrm{Select}(e_{t,:}, \mathcal{U}_t; b_t), \qquad r_t = \mathrm{softmax}\left(\frac{q_t K_{\mathcal{I}_t}^\top}{\sqrt{d}}\right) V_{\mathcal{I}_t}$$

- **Structure:** prescribe a mask or instantiate a profiled pattern.
- **Self-routing:** reuse backbone geometry, pooled keys, or summaries.
- **Auxiliary proxy:** learn a separate address space for selection.

## Cost accounting

Evidence + index scan + Top-$k$ + KV gather + readout + refresh.

Paper, Sec. 4, Eqs. (8)–(9), PDF pp. 15–26. $e$: selection scores; $\mathcal{U}$: available units; $b_t$: access budget. Cost list is a reading synthesis.

# QSA shortens the index scan by pooling before scoring

| DSA | MSA | QSA |
| --- | --- | --- |
| Token index keys | Token index keys | Token index keys |
| ↓ | ↓ | ↓ |
| Token scores | Token scores | Average-pool keys |
| ↓ | ↓ | ↓ |
| Weighted head fusion | Block max-pooling | Micro-block scores |
| ↓ | ↓ | ↓ |
| Shared token Top-$k$ | Block Top-$k$ per group | Shared block Top-$k$ |

DSA, MSA, and QSA retrieve original-token content.

**CSA additionally compresses the global content itself.**

Redrawn schematically from Paper, Fig. 7 and Eqs. (29)–(32), PDF pp. 21–24. Local/current-block inclusion omitted from the schematic.

# Content, addresses, and selected indices can refresh separately

| CSA2 mode | Global KV / index keys | Selected support |
|---|---|---|
| Full | Fresh | Full index scan and new Top-$k$ |
| Reindex | Reuse from Full | New selection within shared candidate pool |
| Reuse | Reuse from Full | Inherit latest selection |

All modes still form their own main query and local-window KVs.

- **IndexCache:** reuse indices while layers retain their own content.
- **Temporal reuse:** recall prior supports, then rerank for the new query.

Paper, Sec. 4.4, Eqs. (33)–(37), PDF pp. 24–26. CSA2 candidate-pool reuse described for decoding.

# Linear attention changes what is stored

For a separable kernel $K(q,k) = \phi(q)^\top \phi(k)$:

$$S_t = S_{t-1} + \phi(k_t) v_t^\top, \qquad z_t = z_{t-1} + \phi(k_t)$$

$$o_t = \frac{\phi(q_t)^\top \widehat{S}_t}{\phi(q_t)^\top \widehat{z}_t + \varepsilon}$$

- $S_t \in \mathbb{R}^{d_k \times d_v}$ compresses key–value associations.
- Fixed dimensions give context-independent persistent state.
- Parallel/chunkwise training can compute the recurrence efficiently.

Paper, Eqs. (38)–(39), PDF p. 27. Hats denote the visible read state; normalization requires an appropriate kernel. Later variants may use unnormalized readout.

# Delta updates correct the association already stored

## Residual-dependent write

$$S_t = S_{t-1} + \beta_t k_t \left(v_t^\top - k_t^\top S_{t-1}\right)$$

**Additive update**

Writes the current value regardless of the old prediction.

Repeated associations can accumulate and interfere.

**Delta update**

Fits the current key–value pair.

GDN adds scalar decay; KDA adds channel-wise decay before correction.

With $\|k\| = 1$ and $\beta = 1$, values $1 \to 3$ at the same key give:
additive readout 4; Delta readout 3 *(teaching example)*.

Paper, Sec. 5.1, Tables 7–8, PDF pp. 27–32. Better updates still operate within finite capacity.

# More memory can mean more rows or more time intervals

## Capacity expansion

**SSE:** grouped state rows with sparse group activation.

**SDM:** flat rows with separate read and write routers.

Total stored capacity and active work are different budgets.

## Temporal expansion

**Log-Linear:** $O(\log t)$ multiscale states.

**DLA:** content-based segments, $M_t \leq K$.

**MHLA:** fixed-block summaries; count may grow.

Routing must align what gets written with what future queries can recover.

Paper, Secs. 5.2–5.3, Tables 9–10, Fig. 8, PDF pp. 32–35.

# Selective SSMs make state dynamics depend on the input

## Time-invariant kernel

$$h_t = \bar{A} h_{t-1} + \bar{B} x_t, \qquad r_t = C h_t$$

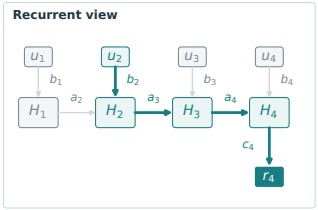
From $h_0 = 0$: $r_t = \sum_{i \le t} C \bar{A}^{t-i} \bar{B} x_i$.

- S4/S4D structure the transition; S5 uses a joint MIMO system.
- Mamba generates $\Delta_t, B_t, C_t$ from the current input; $\bar{A}_t = \exp(\Delta_t A)$.
- Content-dependent controls require selective sequence computation; generator weights stay fixed at inference.

Paper, Sec. 6.1, Eqs. (55)–(56), PDF pp. 37–38. $h_t$: recurrent state; $x_t$: input; $r_t$: readout.

# SSD gives recurrent and matrix views of the same kernel

$$H_t = a_t H_{t-1} + b_t u_t^\top, \qquad r_t = H_t^\top c_t$$




$$r_t = \sum_{i \le t} \underbrace{(c_t^\top b_i)}_{\text{read–write match}} \underbrace{\left( \prod_{j=i+1}^{t} a_j \right)}_{\text{intervening decay}} u_i$$

Paper, Fig. 11, Eqs. (57)–(58), PDF pp. 38–39. $H_t \in \mathbb{R}^{N \times P}$; $b_t, c_t \in \mathbb{R}^N$; $u_t \in \mathbb{R}^P$; $a_t$ is scalar per head.

# Hybrid architecture chooses where capabilities meet

| Granularity | Decision | Example |
|---|---|---|
| Layer | Place different mixers along depth | Jamba, Samba; conversion/search |
| Head | Allocate head/channel capacity | Hymba, DuoAttention, HydraHead |
| Branch | Fuse identifiable memory paths | Infini-attention, DART, Titans-MAG |
| Token | Choose storage or operation per token | LoLCATs, LoLA, NAtS-L |

The granularities can coexist. A hybrid is a composition of mechanisms or access regimes, rather than another memory operator.

Paper, Sec. 7, Fig. 13 and Tables 13–16, PDF pp. 42–48.

# Hybrid records rise within the curated architecture inventory

| Period | Records | Single-family | Hybrid | Cross-layer |
|---|---|---|---|---|
| 2022–2023 | 6 | 6 | 0 | 0 |
| 2024–2025 | 25 | 16 | 9 (36%) | 0 |
| 2026 to Sep 22 | 27 | 10 | 17 (63%) | 5 (19%) |

- 59 release records, 14 lineages; 58 records are classifiable.
- Kimi k1.5 is excluded from percentages because its attention is undisclosed.
- Cross-layer reuse overlaps with the other labels: **do not add the columns**.

Paper, Table 17, PDF p. 50; Table S1, pp. 57–58. Curated release counts, not market share or controlled performance measurements.

# The frozen frontier sample retains diverse explicit paths

| Selected endpoint | Documented attention structure |
|---|---|
| K2-Horizon-375B-A23B | Full GQA |
| GLM-5.3 (max) | MLA-based DSA + index sharing |
| Qwen3.8-Flash-Next | 3 GDN : 1 QSA |
| Kimi K3 (max) | 3 KDA : 1 Gated MLA |
| DeepSeek-V4.1-Flash (max) | CSA2 + local window; cross-layer reuse |

All 11 selected architectures retain an explicit token-retrieval path.
That path may be local; remote content may already be compressed.

Selected rows from Paper, Table 18, PDF p. 50. Artificial Analysis v4.3.2 snapshot: Sep 22, 2026. Describes coexistence; does not isolate causal effects of attention design.

# Network depth governs both placement and persistence

## Capability placement

Which layer performs explicit retrieval, sparse access, or recurrent propagation?

Layer-wise hybrids coordinate through the hidden-state stream.

## Artifact persistence

Which KVs, index keys, or selected supports survive into later layers?

Reuse schedules determine when each object is refreshed.

The inventory's five cross-layer records represent three base patterns across GLM, DeepSeek, and LongCat.

Paper, Secs. 8.2 and 9.2, PDF pp. 49–50 and 52–53. Cross-layer reuse is a separate, overlapping architectural attribute.

# Hypothesis: route persistent memory across four coordinates

$$\mathcal{A}_t = \mathcal{T}_t \times \mathcal{L} \times \mathcal{S} \times \mathcal{G}, \qquad a = (\tau, \ell, s, g)$$

| | |
|---|---|
| Temporal scope $\tau$ | token, segment, long interval |
| Depth $\ell$ | producer or maintenance stage |
| Substrate $s$ | explicit KV, associative state, SSM state |
| Granularity $g$ | exact entries or coarse summaries |

**Sparse Write:** decide where information is retained.

**Sparse Read:** decide which stored information serves a query.

Author hypothesis, Sec. 9.3, Eqs. (68)–(70), PDF pp. 53–54. A logical address space, not a required Cartesian tensor or an evaluated new model.

# Four proposed steps make the hypothesis testable

1. **Choose representations by workload:** exact KV for precision-sensitive regions; summaries and indexes elsewhere.
2. **Separate construction from retrieval:** lightweight memory construction, then coarse-to-fine query-time access.
3. **Control lifecycle incrementally:** reuse, refresh, transform, or invalidate affected artifacts.
4. **Route memory and compute together:** start with a finite menu of memory, depth, and write-back policies.

Write decisions have delayed consequences: future queries can only retrieve what the memory system preserved.

Author proposals, Sec. 9.3.4, PDF pp. 54–55. The complete multidimensional system remains unvalidated.

# The survey is a design map for quality–cost trade-offs

## What the paper contributes

A shared vocabulary, a mechanism taxonomy, an architecture inventory, and a testable research hypothesis.

- Explicit memory preserves selectable units; recurrent memory compresses history into state.
- Hybrid placement and artifact reuse make depth part of memory management.
- Compare retrieval quality, persistent memory, prefill, decoding, and routing under matched budgets.

Adoption counts do not establish an attention mechanism's performance advantage.

Paper, Secs. 8–10, PDF pp. 48–56. Main talk ends here; following slides are technical backup.

# Backup: routing scores can also control readout

HiLS approximates the attention mass of chunk $u$:

$$Z_{t,u} = \sum_{j \in \mathcal{B}_u} e^{s_{t,j}}, \qquad \widehat{s}_{t,u} = \frac{q_t (k'_u)^\top}{\sqrt{d}} + H(\alpha_u)$$

$$k'_u = \sum_j \alpha_{u,j} k_j, \quad \widehat{Z}_{t,u} = e^{\widehat{s}_{t,u}}, \quad w_{t,j} = \frac{e^{s_{t,j}}}{Z_{t,c(j)}} \frac{\widehat{Z}_{t,c(j)}}{Z_{t,\mathrm{loc}} + \sum_{u \in \mathcal{S}_t} \widehat{Z}_{t,u}}$$

- Exact within-chunk weights; surrogate masses across selected chunks.
- The surrogate participates in continuous forward weights, allowing language-model loss to train the summaries.

Paper, Eqs. (23)–(28), PDF p. 20. $\alpha_u$: landmark-induced distribution; $H$: entropy; $c(j)$: chunk of token $j$; $\mathcal{S}_t$: selected distant chunks.

# Backup: richer updates decouple retention, erasure, and writing

Let $S \in \mathbb{R}^{d_k \times d_v}$ and $D_t = \mathrm{Diag}(\alpha_t)$.

$$\text{KDA:} \quad S_t = (I - \beta_t k_t k_t^\top) D_t S_{t-1} + \beta_t k_t v_t^\top$$

$$\text{GDN2:} \quad S_t = [I - k_t (b_t \odot k_t)^\top] D_t S_{t-1} + k_t (w_t \odot v_t)^\top$$

$$\text{EDA:} \quad S_t = (I - \beta_t k_t k_t^\top)(I - \gamma_t e_t e_t^\top) D_t S_{t-1} + \beta_t k_t v_t^\top$$

- GDN2: separate key-side erase control $b_t$ and value-side write control $w_t$.
- EDA: erase at $e_t$, then correct at $k_t$; operation order matters.

Paper, Table 7, PDF p. 31; Sec. 5.1. Feature generation, normalization, and output gates omitted as in the survey's comparison.

# Backup: Delta correction is one state-optimization step

$$\mathcal{L}_t(S) = \frac{\beta_t}{2}\|S^\top k_t - v_t\|_2^2$$

$$\nabla_S \mathcal{L}_t = \beta_t k_t (k_t^\top S - v_t^\top)$$

$$S - \nabla_S \mathcal{L}_t = (I - \beta_t k_t k_t^\top) S + \beta_t k_t v_t^\top$$

- GDN/KDA optimize this target at a decayed state.
- DeltaProduct applies multiple sequential corrections per token.
- EDA first fits a zero target at an erase address, then fits the value.

Paper, Eq. (51) and Table 8, PDF pp. 30–32. Equivalent objectives for the internal state; pretrained model weights are not the optimization variables here.

# Backup: SSM state capacity and interface width are separate

$$H_t \in \mathbb{R}^{N\times P}, \quad C_t^{\text{multi}} \in \mathbb{R}^{N\times R}, \quad Y_t = H_t^\top C_t^{\text{multi}} \in \mathbb{R}^{P\times R}$$

- Larger $N, P$ increase retained state; more directions $R$ enrich readout from the same principal state.
- Direct multi-direction readout costs $O(NPR)$ arithmetic.
- MIMOMamba uses a different matrix-valued algebraic construction.
- HADES organizes temporal filters; MuonSSM conditions input-injection geometry.

Paper, Sec. 6.2, Eq. (59), Fig. 12, PDF pp. 40–41. More read directions cannot recover information no longer encoded in the state.

# Backup: hybrid allocation policies use different evidence

| Layer allocation | Selection signal |
|---|---|
| Predefined | Architecture designer specifies a schedule |
| Direct diagnosis | Original pretrained layer behavior |
| Alignment-based | Replacement behavior after representation alignment |
| Iterative distillation | KL reduction when restoring a layer in an adapted student |

**Head policies also differ:** offline function selection, input-adaptive mode selection, and depth-dependent head budgets.

Paper, Secs. 7.1–7.2, Tables 13–14, PDF pp. 42–45.

# Backup: branch fusion depends on the interface

## Readout-level fusion: Infini-attention

$$A = \sigma(\beta) \odot A_{\mathrm{mem}} + [1 - \sigma(\beta)] \odot A_{\mathrm{dot}}$$

Combine compressive-memory and local-attention contexts, then apply a shared multi-head output transformation.

## Output-level fusion: Titans-MAG

$$o_t^{\mathrm{MAG}} = \mathrm{Gate}(o_t^{\mathrm{SWA}}, o_t^{\mathrm{mem}})$$

Each branch first forms its own output; fusion defines the hybrid output.

Paper, Sec. 7.3, Eqs. (61)–(65), Table 15, PDF p. 46. $A_{\mathrm{mem}}$: compressed-memory readout; $A_{\mathrm{dot}}$: local Softmax readout.

# Backup: token routing determines memory fidelity over time

- **Temporal boundary:** LoLCATs moves expired local tokens into a linear state; NHA uses long-term slots.
- **Content score:** LoLA uses self-recall error; STILL uses self-saliency to retain selected historical KVs.
- **Learned operation:** NAtS-L chooses Softmax or GDN at token/chunk granularity.

$$z_i \in \{\mathrm{KV}_{\mathrm{recent}}, \mathrm{KV}_{\mathrm{salient}}, S_{\mathrm{compressed}}\}$$

The storage decision is made before all future queries are known.

Paper, Sec. 7.4, Table 16 and Eqs. (66)–(67), PDF p. 47. $z_i$: architecture-level allocation decision.

# Backup: frozen endpoint inventory (1/2)

| Endpoint | Attention structure |
|---|---|
| MiMo-V2.6-Pro | 60 SWA-GQA + 10 global-GQA layers |
| DeepSeek-V4.1-Flash (max) | CSA2 + SWA; cross-layer KV/index/support reuse |
| K2-Horizon-375B-A23B | Full GQA |
| GLM-5.3-Flash | 34 KDA + 11 compressed-indexer DSA layers |
| GLM-5.3 (max) | MLA-based DSA; IndexShare/IndexCache |
| Qwen3.8-Flash-Next | 3 GDN : 1 QSA |

Paper, Table 18, PDF p. 50. Artificial Analysis Intelligence Index v4.3.2 snapshot captured Sep 22, 2026. Rows describe endpoint architectures, not a mechanism ranking.

# Backup: frozen endpoint inventory (2/2)

| Endpoint | Attention structure |
|---|---|
| DeepSeek-V4 Pro 0813 (max) | CSA/HCA layer-wise hybrid |
| Qwen3.8-2.4T-A95B | 3 GDN : 1 gated full-GQA layer |
| Qwen3.8-27B (xhigh) | 3 GDN : 1 gated full-GQA layer |
| Kimi K3 (max) | 3 KDA : 1 Gated MLA |
| MiniMax-M3 | 3 dense-attention + 57 MSA layers |

Reasoning settings describe endpoints; they are not separate architectures. The paper does not report index scores in this table.

Paper, Table 18 and Sec. 8.3, PDF p. 50. Snapshot date: Sep 22, 2026.

# Backup: source and reading route

**Primary source**

Zhentao Tan, Jingyi Shen, Yanbo Li, Yao Liu, Yue Wu, Jieping Ye.

*The Evolution of Attention in Large Language Models: Mechanisms, Trade-offs, and Emerging Trends.*

arXiv:2609.39661v1, September 30, 2026.

- Framework: Sec. 2; explicit/sparse memory: Secs. 3–4.
- State updates and SSMs: Secs. 5–6; composition: Sec. 7.
- Architecture evidence: Sec. 8 and Appendix A.
- Synthesis and proposals: Sec. 9; conclusion: Sec. 10.

All mechanism and architecture claims in this presentation are attributed to the survey. Its cited primary literature is listed in the paper, PDF pp. 59–67.