

method	memory at step t , and how step t changes it	what the change adds
Softmax attention Vaswani et al. (2017)	the cache $K_{1:t}, V_{1:t}$: $2td$ numbers	every past token stays available; the memory grows with t
this post Linear attention Katharopoulos et al. (2020)	$S_t = S_{t-1} + \phi(k_t) v_t^\top$	a fixed $r \times d$ memory; pairs are only added, nothing leaves
Delta rule (DeltaNet) Schlag et al. (2021); Yang et al. (2024b)	$S_t = (I - \beta_t k_t k_t^\top) S_{t-1} + \beta_t k_t v_t^\top$	erases what k_t reads before it writes v_t ; trains in chunks
Decay (Mamba-2) Dao and Gu (2024)	$S_t = \alpha_t S_{t-1} + k_t v_t^\top$	forgets every pair at one rate, $\alpha_t \in [0, 1]$
next Gated DeltaNet Yang, Kautz and Hatamizadeh (2025)	$S_t = \alpha_t (I - \beta_t k_t k_t^\top) S_{t-1} + \beta_t k_t v_t^\top$	both rows above: fast erasure and targeted edits
Kimi Delta Attention (KDA) Kimi Team (2025)	$S_t = (I - \beta_t k_t k_t^\top) \text{Diag}(\alpha_t) S_{t-1} + \beta_t k_t v_t^\top$	one forget rate for each feature channel
then Hybrid (Kimi Linear) Kimi Team (2025)	three KDA layers for each full-attention layer	the full-attention layers see every past token; KV cache up to 75% smaller

Linear attention reads $o_t = S_t^\top \phi(q_t) / z_t^\top \phi(q_t)$. From the delta rule on, ϕ is folded into q_t and k_t , and the recurrent rows read $o_t = S_t^\top q_t$. The highlighted term is the main change in each row. In the delta-rule rows, keys have unit length and $\beta_t \in [0, 1]$ is a write strength; the decay row folds its input scale into k_t .