Nectar

Neural Estimation of Cached-Token Attention via Regression

Paper by João Monteiro, Michal Klein,
Pierre Ablin, and Marco Cuturi (Apple)

Research paper discussion ▪ 8 October 2026
arXiv:2605.09778v2, revised 8 July 2026

Compile a fixed document's attention behavior into small networks.

A fixed document is expensive to read repeatedly

Serving the same book, manual, or knowledge source

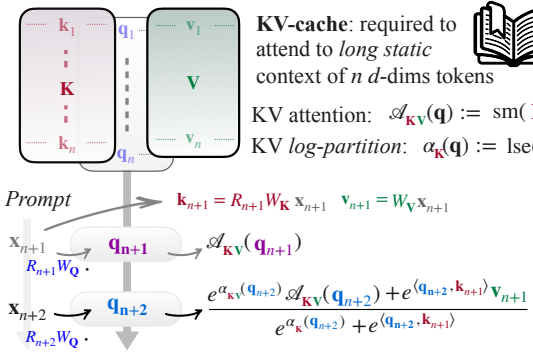
Every new query reads the document's cached keys and values.

$$\text{Document KV state} = 2nLHd \text{ scalar entries}$$

- n: document tokens; L: layers; H: KV heads; d: head size.
- Great Gatsby, about 61K tokens, on Qwen3-4B:
  about **9 GB of KV state** versus about 8 GB of model weights.
- Opportunity: pay once to fit the document's query response, then reuse the fitted module across requests.

Paper, Sec. 1, PDF p. 1. Storage example uses bfloat16.

Nectar stores the function induced by the document



KV-cache: required to attend to *long static* context of n d-dims tokens

KV attention: $\mathscr{A}_{KV}(\mathbf{q}) := \mathrm{sm}(\mathbf{K}\,\mathbf{q})\mathbf{V}$

KV *log-partition*: $\alpha_K(\mathbf{q}) := \mathrm{lse}(\mathbf{K}\,\mathbf{q})$

NECTAR: *replace* costly KV-attention operator using two small NNs.

Parameter size: $|\theta| \ll 2nd$

$\approx A_\theta(\mathbf{q})$ $\approx a_\theta(\mathbf{q})$ ~~$\mathscr{A}_{KV}$~~

Learn θ using regression & distillation losses

Per layer/KV-head	Memory	Inference for T tok.
KV-cache	$2nd$ ✗	$T2nd + d\frac{T(T-1)}{2}$ ✗
NECTAR	$\lvert\theta\rvert$ ✓	$\lvert\theta\rvert T + d\frac{T(T-1)}{2}$ ✓

One pair per layer and KV head

Predict the document's attention output and its log-normalizer.

Paper, Fig. 1, PDF p. 2. Original figure. Base-model weights remain frozen.

Two functions separate retrieved content from mass

For a fixed document, $\mathbf{K}, \mathbf{V} \in \mathbb{R}^{n \times d}$ and a rotated query $\mathbf{q} \in \mathbb{R}^d$, let $s_j = \mathbf{q}^\top \mathbf{k}_j / \sqrt{d}$.

$$\underbrace{\alpha(\mathbf{q}) = \log \sum_{j=1}^{n} e^{s_j}}_{\text{log attention mass}} \qquad \underbrace{\mathcal{A}(\mathbf{q}) = \frac{\sum_{j=1}^{n} e^{s_j} \mathbf{v}_j}{\sum_{j=1}^{n} e^{s_j}}}_{\text{retrieved content}}$$

- Score network: $a_\theta(\mathbf{q}) \approx \alpha(\mathbf{q})$, a scalar.
- Target network: $A_\theta(\mathbf{q}) \approx \mathcal{A}(\mathbf{q})$, a d-vector.
- GQA: query heads sharing a KV head also share the module; each query is still evaluated separately.

Paper, Sec. 3 and Eq. (3.1), PDF p. 4. Layer/head indices suppressed.

The score makes exact local attention composable

For visible local tokens, let $Z_{\text{loc}} = \sum_i e^{s_i}$, $\mathbf{N}_{\text{loc}} = \sum_i e^{s_i}\mathbf{v}_i$, and $\mathcal{A}_{\text{loc}} = \mathbf{N}_{\text{loc}}/Z_{\text{loc}}$ when $Z_{\text{loc}} > 0$.

$$\text{ATT}_{\text{full}} = \frac{e^{\alpha}\mathcal{A} + \mathbf{N}_{\text{loc}}}{e^{\alpha} + Z_{\text{loc}}} \quad \longrightarrow \quad \widehat{\text{ATT}} = \frac{e^{a_\theta} A_\theta + \mathbf{N}_{\text{loc}}}{e^{a_\theta} + Z_{\text{loc}}}$$

Equivalent gate, when $Z_{\text{loc}} > 0$

$g = \text{sigmoid}(a_\theta - \log Z_{\text{loc}}), \quad \widehat{\text{ATT}} = g\mathcal{A}_\theta + (1-g)\mathcal{A}_{\text{loc}}.$

- Splitting the full softmax sum is exact; learned functions introduce error.
- Insert one virtual logit/value pair alongside visible local tokens.
- Local instruction and generated-token KV state is retained.

Paper, Eqs. (3.4)–(3.5), PDF p. 5. Gate form is a teaching derivation.

The representation and the loss are separate choices

Quadrature: learned KV pairs

$\mathbf{W}, \mathbf{Z} \in \mathbb{R}^{p \times d}$, $p \ll n$

$$A_\theta(\mathbf{q}) = \text{softmax}(\mathbf{W}\mathbf{q}/\sqrt{d})^\top \mathbf{Z}$$

- $2pd$ parameters; initialize from the first p cached pairs.
- Score follows analytically from $\mathbf{W}$.
- With pure KL: Cartridges baseline under this paper's data recipe.

MLP: direct function regression

Separate scalar and vector outputs

$$\mathbf{h}_{k+1} = \text{SiLU}(\mathbf{V}_k\mathbf{q} + \mathbf{U}_k\mathbf{h}_k + \mathbf{b}_k)$$

- Re-inject the query at intermediate depths.
- Optional shared backbone, then score and target heads.
- (D_b, D_s, D_t) specifies their depths; width fits the budget.

Paper, Sec. 3.3, Eq. (3.6), PDF pp. 5–6; App. A.3, p. 13.

Train internal fidelity and final behavior together

$$\mathcal{L}_{\mathcal{A}} = \mathbb{E} \sum_{\ell,h,h'} \left\| A_\theta^{\ell,h}(\mathbf{q}_i^{\ell,h'}) - \mathcal{A}^{\ell,h}(\mathbf{q}_i^{\ell,h'}) \right\|_2^2$$

$$\mathcal{L} = \underbrace{\lambda_\alpha \mathcal{L}_\alpha + \lambda_{\mathcal{A}} \mathcal{L}_{\mathcal{A}}}_{\mathcal{L}_{\text{reg}}} + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}$$

- $\mathcal{L}_\alpha$: analogous scalar score regression.
- KL matches the full-cache teacher's next-token distribution under teacher forcing.
- MLP: $(\lambda_\alpha, \lambda_{\mathcal{A}}) = (0.1, 1)$; Quadrature: $(0, 1)$. Pure KL sets both to zero.
- Only module parameters change; queries come from task-relevant data.

Paper, Sec. 3.1, Eqs. (3.2)–(3.3), PDF pp. 4–5.

Self-study defines the queries the module learns

1. Split each document into overlapping chunks of about 2,000 words.
2. Qwen3-8B generates about 10 pairs per chunk:
   5 QA, 1 summary, 2 quoting, 2 paraphrasing.
3. Qwen2.5-7B-Instruct-1M re-answers using the full document; discard low-consistency pairs (confidence below 0.7).
4. Random 80/20 split; cache the full model's attention targets.

Coverage is part of the approximation problem

Train on model-generated queries for one document; the training instruction-plus-response length is capped at 196 tokens.

Paper, Apps. A.1–A.4, PDF pp. 12–14; App. I, pp. 24–25, 34–36.

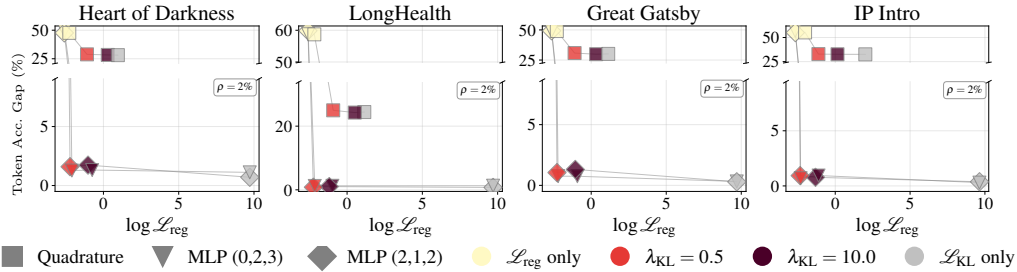
The evaluation has three different scopes

Question	Models	Contexts / measurements
Attention / token fit	Qwen3: 1.7B, 4B, 8B; Qwen2.5-7B-1M	Five documents, 40K–122K tokens
Free-form generation	Qwen2.5-7B-1M	Three documents; QA, summary, quoting, paraphrasing
Serving cost	Qwen3-1.7B and 4B	TTFT, decode speed, peak GPU memory

- Qwen3 uses YaRN 4×; Qwen2.5-7B-1M needs no extension.
- $\rho = |\theta|/(2nd)$: module parameters relative to document KV entries.

Paper, Sec. 4, PDF pp. 6–9; App. D, pp. 18–19.

Internal error and token accuracy can disagree



- Qwen2.5-7B-1M, $\rho = 2\%$; lower and left are desirable.
- Shape: architecture. Color: loss. Vertical axes contain breaks.
- Pure regression can have low internal error and large token gaps; pure KL can do the reverse. Mixed MLPs balance both.

Paper, Fig. 4, PDF p. 7. Original crop; interpretation distinguishes loss regimes.

Good surrogate metrics do not ensure usable rollouts

First verify that the teacher can generate

All tested YaRN-Qwen3 models failed the long-context generation test, including the full-cache teacher. YaRN is the authors' suspected cause.

On Qwen2.5-7B-1M at $\rho = 2\%$, MLP generation gaps are:

Objective	Heart of Darkness	LongHealth	Great Gatsby
Pure regression	2.22	2.22	2.36
Regression + $0.01\,\mathrm{KL}$	0.18	0.39	0.26
Pure KL	0.98	1.04	0.62

Paper, Sec. 4.3, PDF p. 8; Table 14, p. 26. $\Delta_s \downarrow$, 1–5 judge scale.

Generation is measured by semantic similarity

$$\Delta_s = \bar{s}_{\text{base}} - \bar{s}_{\text{Nectar}} \qquad \text{(lower is better)}$$

- Three documents, 500 held-out instructions per document.
- GPT-5.2 rates responses against synthetic reference answers on a 1–5 scale.
- Sampling: temperature 0.7, top-$p = 0.8$; at most 64 tokens for QA and 196 for other tasks.
- Even quoting is judged semantically, with minor wording changes allowed.

Selection matters

Capacity results select the best configuration per document and capacity, then average over three documents; CIs describe these aggregation units.

Paper, Sec. 4.3, PDF p. 9; Apps. F and H, pp. 21, 23–24, 32–33.

More capacity brings MLP close to full-cache quality

	$\rho = 2\%$	$\rho = 5\%$	$\rho = 10\%$
MLP	0.28 ± 0.26	-0.02 ± 0.39	-0.14 ± 0.34
Quadrature	2.25 ± 0.22	2.23 ± 0.21	2.22 ± 0.26

- Random held-out questions; Qwen2.5-7B-1M; $\Delta_s \downarrow$.
- The MLP mean improves across capacities; the same trend is not strictly monotonic within every document.
- At 10%, the 95% CI crosses zero: evidence for near-baseline quality, with an uncertain small mean advantage.
- Quadrature remains poor under this paper's data and training recipe.

Paper, Table 3, PDF p. 9; per-document values in Table 13, p. 20.

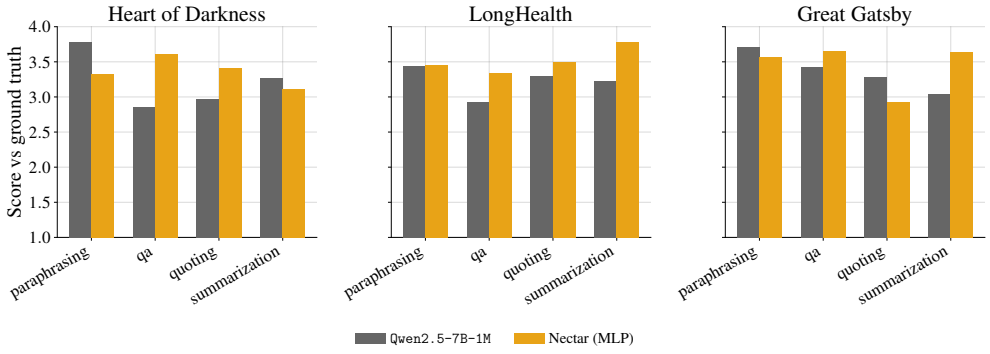
A small distillation weight helps on average

MLP objective	Random-split $\Delta_s \downarrow$
Pure KL	0.36 ± 0.33
Regression $+\ 0.01\,\mathrm{KL}$	0.25 ± 0.13
Regression $+\ 2\,\mathrm{KL}$	0.28 ± 0.37
Regression $+\ 10\,\mathrm{KL}$	0.53 ± 0.24

- Fix the loss weight; select within each document–capacity cell; average over cells. This differs from capacity-wise selection.
- At higher capacity, pure KL wins some cells; mixed training wins others. One loss weight does not dominate every setting.

Paper, Table 4, PDF p. 9; detailed cells in Table 14, p. 26.

Quality varies with the document and task



- Absolute judge scores: higher is better. Gray: full cache; gold: MLP.
- Best overall run per document: QA gap -0.46; paraphrasing gap $+0.19$. Task-level CIs are wide.
- Semantic match can coexist with unsupported extra details in individual responses.

Paper, Fig. 17, PDF p. 25; Table 5, p. 10; example in Table 2, p. 9.

Dissimilar questions expose a substantial gap

Challenge: select the 500 test instructions per document with the lowest maximum embedding similarity to any training instruction.

MLP $\Delta_s \downarrow$	$\rho = 2\%$	$\rho = 5\%$	$\rho = 10\%$
Random	0.28 ± 0.26	-0.02 ± 0.39	-0.14 ± 0.34
Challenge	1.40 ± 0.54	1.28 ± 0.49	1.18 ± 0.43

- Same documents; a different query distribution.
- Challenge quoting gap: 1.98 in the best-run task breakdown.
- Quadrature is worse (about 2.78), but MLP no longer matches full cache.

Paper, App. F, PDF pp. 21–22; Tables 19 and 21, p. 27. Encoder: all-MiniLM-L6-v2.

Spend capacity where it improves the final output

Qwen3-1.7B, MLP, $\rho \approx 2\%$, $\lambda_{\mathrm{KL}} = 0.01$:

Three-document mean	Uniform	Weighted	Change
Score MSE	0.056	0.074	Worse
Target MSE	0.256	0.224	Better
Token-accuracy gap (pp)	0.76	0.60	−0.16 pp

- Layer groups: $[0, 9)$, $[9, 15)$, $[15, 27)$, $[27, 28)$.
- Multipliers $(1, 2, 5, 2)$, normalized to preserve total capacity.
- More middle-to-late capacity helps target fidelity and token accuracy, despite a slightly worse score fit.

Paper, App. C.1, Table 10, PDF p. 16. pp = percentage points.

The large storage reduction applies to document state

At about 62K tokens, bfloat16 storage:

Model	Weights	Full KV	2% module	10% module
Qwen3-4B	8.04 GB	9.14 GB	183 MB	914 MB
Qwen2.5-7B-1M	14.14 GB	3.56 GB	71 MB	355 MB

- 2% means about 50× less *static document state*.
- Model weights, local-token KV, and working memory remain.
- A fixed fitted module has no document-length loop at inference.
- Across documents at fixed ρ, its budget $|\theta| = \rho\, 2nd$ still grows with n.

Paper, Table 9, PDF p. 15; Fig. 1, p. 2. Last distinction is a complexity interpretation.

Serving speed improves under an asymmetric prefill setup

At 122,466 document tokens; each cell is Base → Nectar:

Model	TTFT (ms)	Decode (tok/s)	Peak memory (GB)
Qwen3-1.7B	299,900 → 29.3	4.4 → 37.7	60.4 → 8.2
Qwen3-4B	557,600 → 50.1	1.5 → 15.7	84.2 → 17.1

- Decode improves by 8.6× and 10.5× in these rows.
- Base includes full-document prefill; Nectar uses a trained module. Offline work is excluded; prefix-cache reuse is not the baseline.
- These YaRN-Qwen3 models failed the generation-quality test. Usable quality and speed are not jointly established in this table.

Paper, App. D, Table 12, PDF pp. 18–19; quality failure in Sec. 4.3, p. 8.

The trade-off favors stable, frequently queried documents

Offline cost must be amortized

Qwen2.5-7B-1M, mixed training on an 8-GPU node:
16.3 h at 2% / 1M samples; 24.0 h at 10% / 1.5M samples.
Reported times already subtract one hour of loading and compilation.

- A module is tied to its document and base model.
- Query coverage matters: challenge prompts lose much more quality.
- Training and reported generation are short after the document; long conversation drift remains an open deployment question.
- Needed comparison: quality, speed, and amortization on the same usable model against a reused prefix KV-cache.

Paper, Table 8, PDF p. 15; Sec. 5, p. 10; Apps. F and H. Final item: reading analysis.

Three ideas to take into further research

1. **Content plus mass is a useful interface.**
   The log-normalizer makes an approximate document response composable with exact local attention.

2. **Learn the query distribution as carefully as the module.**
   The same fixed document supports both near-baseline random-split quality and a substantial challenge-split gap.

3. **Validate rollouts and costs on one coherent setup.**
   Internal regression, token agreement, semantic generation, and serving efficiency measure different properties.

Synthesis of Secs. 3–5 and Apps. C–F. Proposed deployment interpretation, not an additional experiment.

Reference and discussion route

João Monteiro, Michal Klein, Pierre Ablin, Marco Cuturi.

Nectar: Neural Estimation of Cached-Token Attention via Regression.
arXiv:2605.09778v2, revised 8 July 2026.

https://arxiv.org/abs/2605.09778v2

- Mechanism: Eqs. (3.1)–(3.6), PDF pp. 4–6.
- Random generation: Tables 3–6, PDF pp. 9–10.
- Pure regression and challenge: Tables 14, 19–23, pp. 26–28.
- Costs: Tables 8–9 and 12, pp. 15 and 18.

Original figures reproduced from the paper. All page locators refer to the downloaded v2 PDF.

Backup: target capacity matters under mixed supervision

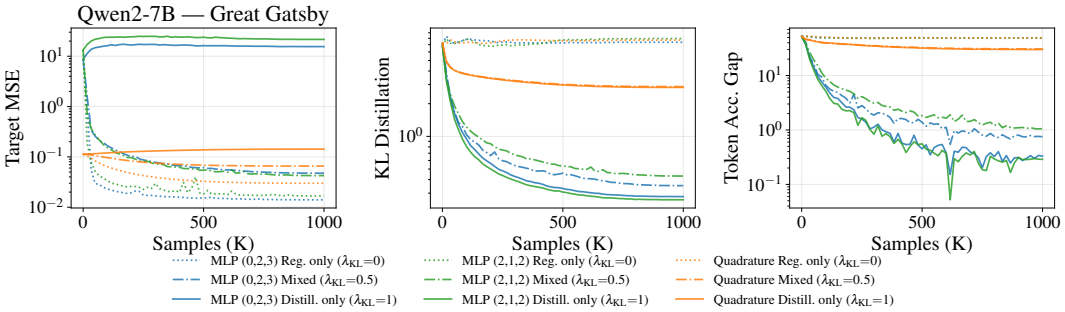
Total $\rho = 10\%$; score/target splits are percentages of full KV state.

Mean token-accuracy gap (pp)	1% / 9%	2.5% / 7.5%
Pure KL ($\lambda_{\mathrm{KL}} = 1$)	0.24	0.20
Mixed ($\lambda_{\mathrm{KL}} = 2$)	0.34	0.47

- With direct score supervision, assigning more capacity to the target reduces the mean token gap in this ablation.
- Pure KL shows a small difference in the opposite direction.
- Layer multipliers here are $(1, 2, 8, 12)$, different from the Qwen3-1.7B uniform/weighted experiment.

Paper, App. C.2, Table 11, PDF pp. 16–17; model labeled Qwen2-7B by the authors.

Backup: pure KL can improve output while MSE rises



- Great Gatsby, $\rho = 2\%$; horizontal axes count training pairs.
- Solid MLP curves: pure distillation. Dotted: pure regression.
- Falling token gap with rising target MSE is visible under pure KL. The internal trajectory need not mimic the teacher's.

Paper, Fig. 9, Great Gatsby row, PDF p. 21. Original crop; paper's model label retained.

Backup: the score connects attention to support functions

With $\alpha(\mathbf{q}) = \log \sum_j \exp(\mathbf{q}^\top \mathbf{k}_j / \sqrt{d})$:

$$\nabla_{\mathbf{q}} \alpha(\mathbf{q}) = \frac{1}{\sqrt{d}} \mathbf{K}^\top \operatorname{softmax}(\mathbf{K}\mathbf{q}/\sqrt{d})$$

$$\lim_{\beta \to \infty} \frac{1}{\beta} \log \sum_j \exp(\beta \mathbf{q}^\top \mathbf{k}_j) = \max_j \mathbf{q}^\top \mathbf{k}_j$$

- Log-sum-exp is convex; the low-temperature limit is a support function.
- This motivates the input-skip MLP design from amortized MIPS.
- The score gradient averages *keys*; target attention averages *values*. They differ in general.
- The paper does not impose convexity guarantees on its SiLU MLP.

Paper, App. B, Eqs. (B.1)–(B.2), PDF pp. 15–16; gradient above restores the Sec. 3 scale.

Backup: RTE needs care when aggregating heads

The paper defines the relative transport error as

$$\mathcal{E}_{\mathrm{rel}} = \sum_{\ell,h} \mathbb{E}_{\mathbf{q}} \left[\log \frac{\|A_\theta^{\ell,h}(\mathbf{q}) - \mathcal{A}^{\ell,h}(\mathbf{q})\|_2^2}{\|\mathbf{q} - \mathcal{A}^{\ell,h}(\mathbf{q})\|_2^2} \right].$$

- The denominator measures the error of returning the query itself.
- For one head, exponentiating the expected log ratio gives its geometric-mean error ratio.
- With a sum over heads, exponentiation gives a product of those means. A global geometric mean needs division by the group count.

Interpretation

Compare like-for-like architectures and normalization conventions; RTE is an internal diagnostic, not a generation score.

Paper, App. C.4, Eq. (C.1), PDF p. 17. Aggregation clarification is this reading's derivation.

Backup: document sizes and training budgets

Document	Tokens	Synthetic pairs
Time Machine	40,022	117,111
Heart of Darkness	49,313	117,120
LongHealth	59,031	103,592
Great Gatsby	61,651	114,896
IP Intro	122,466	228,193

- Query length: cache up to 512 tokens; train on at most 196.
- Table 8 uses 1M training samples at 2%, but 1.5M for Qwen2.5-7B-1M at 5% and 10%.
- Base peak learning rate 10^{-4}; warmup/cosine schedule; global gradient clipping at 1.0.

Paper, Table 1, PDF p. 7; Apps. A.1–A.6 and Table 8, pp. 12–15.

Backup: two reporting details affect reproduction

Qwen3-8B layer count and memory

Sec. 4.1 reports 64 layers and Table 9 gives 16.25 GB of document KV. The official Qwen3-8B config has 36 layers, 8 KV heads, and $d = 128$. At 62,000 tokens in bfloat16 this gives about 9.14 GB.

Optimizer specification

The main text says AdamW; App. A.4 says Adam. Weight decay should be resolved against the implementation for reproduction.

Paper, Sec. 4.1, PDF pp. 6–7; App. A.4, p. 14; Table 9, p. 15.

External configuration check, 8 Oct. 2026: Qwen/Qwen3-8B official config.json.