TRAINING-FREE LATENT MEMORY TRANSFER ACROSS HETEROGENEOUS LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-model LLM systems share instructions, evidence, and interaction history across model families, yet each model keeps that context in a private KV cache. Handing it over as text forces the target to prefill the shared context again, repeating computation on every handoff. Direct cache transfer is hard: models differ in tokenization, positions, layer and head layout, and channel bases, and the target's attention weights each cache error in a query-dependent way. A transfer must preserve the target's computation, not merely match cache values. We present TACIT (Target-Attention-Guided Closed-Form Inter-Model Transfer), the first training-free method for KV-cache transfer across heterogeneous model architectures. TACIT aligns source and target tokens and positions, uses target attention sensitivity to select source layers, and fits per-head K/V maps in closed form. One frozen mapper is fitted per source–target direction from paired unlabeled prefills, then reused across requests and tasks so the target continues from transferred memory. Across five model families, 13 cross-family directions, and 560M–32B models, TACIT retains 83.33% of target-native accuracy on six multiple-choice tasks and outperforms all evaluated transfer baselines on every task. Cache handoff averages 6.13 ms, $6.77\times$ faster than these baselines. On 1K–32K prefixes, end-to-end time-to-first-token improves by $1.52\times$–$10.48\times$ while retaining 92.4–95.4% of next-token accuracy. Analysis shows why: attention-guided selection preserves the target's key routing, and sensitivity-weighted fitting preserves its value readout, turning model-private KV state into a reusable latent-memory interface between heterogeneous models.

1 INTRODUCTION

Modern LLM systems increasingly combine models from different families within a single workflow, routing queries between models, passing intermediate results between agents, and refining one model's output with another (Ong et al., 2025; Wu et al., 2024; Wang et al., 2025). Every such handoff must carry shared context, such as instructions, evidence, and interaction history, across a model boundary. Each model, however, keeps that context in its own KV cache, and text is the only interface it shares with the others. Handing context over as text therefore forces the receiving model to prefill it again (fig. 1(a)), a cost that dominates time-to-first-token for long contexts and recurs on every handoff.

Within one model, serving systems reuse KV state aggressively (Kwon et al., 2023; Gim et al., 2024; Zheng et al., 2024), but they assume a shared cache interface, and across model families that assumption fails at every level. Tokenizers segment the same text differently; positional schemes differ (Su et al., 2021; Press et al., 2022); layer counts and KV-head layouts differ (Ainslie et al., 2023); and channel bases are unrelated even where shapes match. Two routes have been explored. Learned KV-cache translators (Fu et al., 2026; Lee et al., 2026; Rossi et al., 2026; Dery et al., 2026) train neural maps from source to target caches by iterative optimization (fig. 1(b)), at the cost of a training run per source–target configuration. Closed-form linear maps (Heo et al., 2026) avoid training, but have only been shown within a model family, where tokenizer and positional scheme are shared. Both routes share one objective: a mapper is judged by how closely its output matches the target's own cache values.

We argue that this objective is misaligned with what the target needs. The target reads its cache through attention, and attention is far from uniform: a few positions and heads receive most of the weight while many receive almost none (Xiao et al., 2024; Zhang et al., 2023). The same K/V error is therefore nearly harmless at one position and decisive at another, and which is which depends on

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

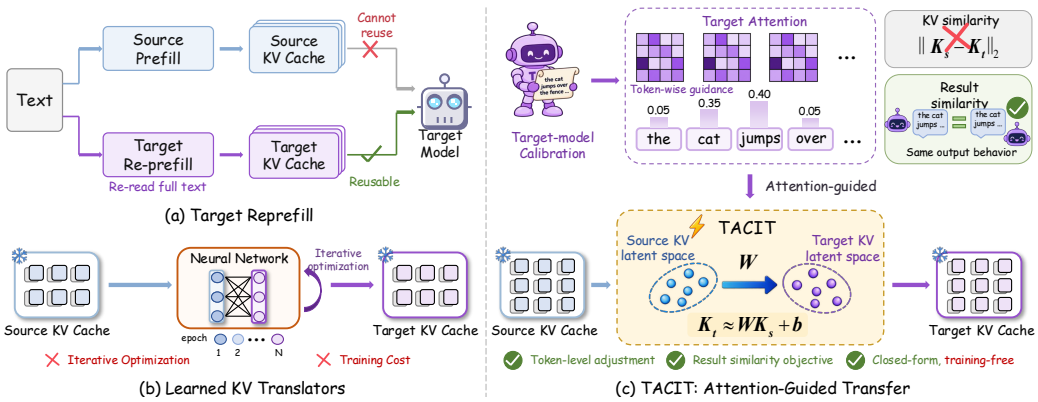

Figure 1: TACIT transfers latent memory across heterogeneous language models through target-attention-guided, closed-form cache conversion, allowing the target to continue from the translated state in its native cache.

the query being processed. Preserving the target's computation thus requires weighting each error by the attention the target will actually pay to it.

Building on this view, we present TACIT (Target-Attention-Guided Closed-Form Inter-Model Transfer), a training-free framework for KV-cache transfer across heterogeneous model architectures (fig. 1(c)). TACIT aligns the source and target token streams and places source positions in the target's positional frame. It then measures how sensitive each target layer's attention is to its keys and values, uses this sensitivity to select the source layers worth reading from, and fits per-head K and V maps in closed form with the same sensitivity as weights. One frozen mapper is fitted per source–target direction from paired unlabeled prefills, in minutes and without backpropagation, then reused across requests and downstream tasks: the target continues directly from the transferred cache as if it had prefilled the context itself. That independently trained networks develop linearly relatable representations (Kornblith et al., 2019; Bansal et al., 2021) suggests why closed-form maps can suffice once they are fit to the right objective. Our contributions are as follows:

- **First training-free KV-cache transfer across heterogeneous architectures.** TACIT transfers caches between models that differ in tokenizer, positional scheme, layer count, KV-head layout, and channel basis, without gradient steps or per-pair training. Frozen mappers cover five model families, 13 directed pairs, and 560M–32B models.
- **Target-attention-guided closed-form fitting.** We formulate transfer as preserving the target's computation rather than matching its cache values: target attention sensitivity decides which source layers are read and how each head's K and V fits are weighted, and every fit is solved in closed form from paired unlabeled prefills.
- **Results and mechanism.** Across six multiple-choice tasks, TACIT retains 83.33% of target-native accuracy and outperforms all four learned translators on every task, with 6.13 ms mean handoff, $6.77\times$ faster than the baselines. On 1K–32K prefixes it delivers $1.52\times$–$10.48\times$ end-to-end TTFT speedups while retaining 92.4–95.4% of next-token accuracy. Analysis shows why: sensitivity-guided layer selection preserves how the target routes keys, and sensitivity-weighted fitting preserves how it reads values.

2 Problem Formulation

A source model may already have prefetched a shared prefix when a downstream target from another family must continue the same workflow. The two models use different token boundaries, positional conventions, layer/head organizations, and channel bases, yielding incompatible cache interfaces across the two models. Effective transfer must preserve the target computation induced by the transferred prefix while eliminating a second target prefill at inference. Accordingly, given frozen source and target models S and T sharing prefix x, the cross-family KV-cache transfer problem is to construct a target-native cache $\widehat{C}_T$ from $C_S(x)$ while retaining the target's continuation behavior. Their tokenizers produce $s = \mathrm{Tok}_S(x) = (s_1, \ldots, s_{n_S})$ and $t = \mathrm{Tok}_T(x) = (t_1, \ldots, t_n)$. Token-end offsets e^S, e^T place both tokenizations in a shared normalized-text coordinate system, while p^S, p^T retain native position IDs.

The source prefill computes the shared context, $C_S(x) = \text{Prefill}_S(s)$. For replay length $0 \le R < n$, let $n_c = n - R$: we transfer $t_{1:n_c}$ and reserve $r = t_{n_c+1:n}$ for native target replay at the transfer boundary. Let $C_T = \text{Prefill}_T(t_{1:n_c})$ be the target-native reference used for mapper construction and evaluation, $\widehat{C}_T = \widehat{C}_T(t_{1:n_c})$ the transferred cache, $q_{1:m}$ the native query/instruction tokens, and $y_{1:o}$ the continuation. A valid directed transfer operator satisfies

$$\widehat{C}_T = \mathcal{T}_{S\to T}\left(C_S(x); e^S, e^T_{1:n_c}, p^S, p^T_{1:n_c}\right), \quad p_T(y_{1:o} \mid \widehat{C}_T, r, q_{1:m}) \approx p_T(y_{1:o} \mid C_T, r, q_{1:m}). \tag{1}$$

This contract couples interface validity and functional fidelity: $\widehat{C}_T$ must inhabit the target-native cache space $\mathcal{C}_T$ and preserve its effect on target attention and continuation. Equal raw K/V errors can induce different target behavior under different queries, attention weights, and value readouts, so target behavior defines the task criterion. The directed operator $\mathcal{T}_{S\to T}$ is frozen once constructed and reused across unseen prefixes and downstream tasks. At inference, S prefills x and T loads $\widehat{C}_T$, replays r, and continues from the transferred memory.

3 METHOD

TACIT organizes transferable memory around a direction-specific frozen mapper (fig. 2). Causal alignment and positional canonicalization establish compatible interfaces; a target-local attention metric then guides source-layer selection and head- and branch-specific closed-form K/V fitting. Paired unlabeled prefills fit this mapper once per direction through closed-form solves.

The central object for a directed pair $S \to T$ is

$$\theta_{S\to T} = \{\mathcal{I}_\ell, W^K_{\ell h}, c^K_{\ell h}, W^V_{\ell h}, c^V_{\ell h}\}_{\ell,h}, \tag{2}$$

where ℓ indexes a target layer, h a target KV head, and $\mathcal{I}_\ell$ selects an ordered tuple of source layers shared by K and V. The affine pairs $(W^Z_{\ell h}, c^Z_{\ell h})$ remain specific to each target layer, KV head, and branch $Z \in \{K, V\}$. Once fitted, the frozen mapper is reused across unseen prefixes and tasks.

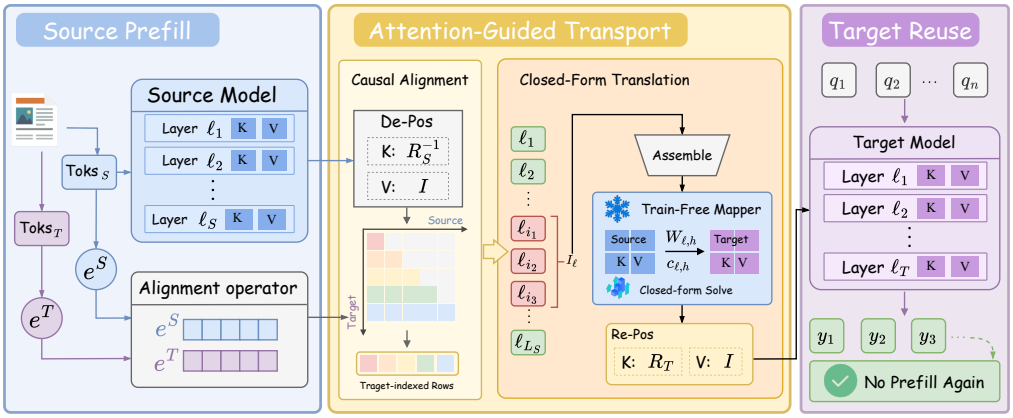

Figure 2: TACIT converts source KV into a target-native cache through causal alignment and attention-guided closed-form fitting, enabling direct reuse of the transferred prefix.

3.1 TOKENIZER-AWARE CAUSAL ALIGNMENT

Raw source and target rows differ because tokenizers place boundaries at different text offsets and cached keys carry model-specific positional transforms. TACIT decouples these mismatches from content translation by aligning de-positioned source states on the target timeline. For key row $K_{j:}$, let $R_j = R(p_j)$ be its invertible positional transform; DePos removes the source transform and RePos later applies the target transform:

$$\text{DePos}(K)_{j:} = K_{j:}R_j^{-1}, \quad \text{RePos}(\bar{K})_{j:} = \bar{K}_{j:}R_j, \quad \text{DePos}(V) = V, \quad \text{RePos}(\bar{V}) = \bar{V}. \tag{3}$$

For RoPE, R_j is the native rotary transform (Su et al., 2021); for ALiBi, $R_j = I$ and positional bias remains in the target attention logits (Press et al., 2022).

After canonicalization, target row $u \in \{1, \dots, n_c\}$ gathers the last source row causally visible at endpoint e_u^T. We call this target-to-source one-hot mapping the alignment operator, $A_x = A_{S \to T}(x) \in \{0,1\}^{n_c \times n_S}$; it places source features on the target timeline while preserving causal visibility. For source layer ι and KV head $h_S \in \mathcal{H}_S$, the complete transport is

$$\begin{aligned} &\pi_x(u) = \max(\{j : e_j^S \le e_u^T\} \cup \{0\}), \quad [A_x]_{uj} = \mathbf{1}\{\pi_x(u) > 0 \wedge j = \pi_x(u)\}, \\ &\widetilde{Z}^S_{\iota h_S} = A_x \, \mathrm{DePos}_{S,\iota}(Z^S_{\iota h_S}), \; F^Z_\ell = \mathrm{Cat}_\ell(\widetilde{Z}^S), \; \widehat{Z}^T_{\ell h} = \mathrm{RePos}_{T,\ell}(F^Z_\ell W^Z_{\ell h} + \mathbf{1} c^Z_{\ell h}), \; Z \in \{K, V\}. \end{aligned} \tag{4}$$

A_x performs a one-hot-or-zero last-causal gather, preserving one source row per target row; Cat_ℓ assembles the selected source layers and heads before affine translation, and RePos restores target positions. The resulting target-indexed rows feed Selection and Fit during construction and are packed into the target-native cache at inference. The main text fixes this alignment and packing convention; boundary cases and positional variants are detailed in section A.2.

3.2 Target-Local Attention Metric

Cross-family conversion errors acquire their functional meaning through the target attention that reads each transferred K/V row. TACIT therefore defines a target-local metric in output space. For a complete calibration span x_{cal}, let $C_{T,\mathrm{cal}} = \mathrm{Prefill}_T(\mathrm{Tok}_T(x_{\mathrm{cal}}))$ and, for a candidate transfer, let $\Delta C = \widehat{C}_{T,\mathrm{cal}} - C_{T,\mathrm{cal}}$. For scaled dot-product attention (Vaswani et al., 2017) at query slot i, target layer ℓ, and target query head g, $\Phi_{i\ell g}$ returns the local attention output before the output projection, with the native query fixed. TACIT measures

$$\mathcal{L}_{\mathrm{attn}}(C_{T,\mathrm{cal}} + \Delta C; C_{T,\mathrm{cal}}) = \sum_{i,\ell,g} \|\Phi_{i\ell g}(C_{T,\mathrm{cal}} + \Delta C) - \Phi_{i\ell g}(C_{T,\mathrm{cal}})\|_2^2 , \tag{5}$$

which measures cache fidelity through the target computation itself.

To obtain a local form that can guide construction, TACIT stacks the maps as Φ_T and linearizes them at $C_{T,\mathrm{cal}}$:

$$\Phi_T(C_{T,\mathrm{cal}} + \Delta C) - \Phi_T(C_{T,\mathrm{cal}}) = J_T \Delta C + \mathcal{O}(\|\Delta C\|_2^2), \qquad \mathcal{L}_{\mathrm{attn}} \approx \underbrace{\Delta C^\top J_T^\top J_T \Delta C}_{\mathcal{L}_{\mathrm{local}}}. \tag{6}$$

The Gauss–Newton pullback (Martens, 2020) assigns larger cost to cache directions that perturb target attention more strongly. Let $a_{ij\ell g}$ be the native attention weight from query i to cache slot j; let $q_{i\ell g}$, $v_{j\ell h(g)}$, and $o_{i\ell g}$ denote the native query, cached value, and attention output; and let $h(g)$ identify the target KV head serving g, including shared KV heads in grouped-query attention (Ainslie et al., 2023). With $\tau_\ell = d_{k,\ell}^{-1/2}$,

$$J^V_{ij\ell g} = a_{ij\ell g} I, \qquad J^K_{ij\ell g} = \tau_\ell a_{ij\ell g} (v_{j\ell h(g)} - o_{i\ell g}) q_{i\ell g}^\top. \tag{7}$$

Values enter the output through direct readout; keys modulate value routing through the softmax. TACIT consequently retains separate K- and V-path metrics.

Closed-form fitting requires separable weights. TACIT therefore measures strict-future ($i > j$) sensitivity on native calibration prefills, matching the causal access pattern of replay and task-suffix queries, and retains the same-slot, same-path blocks for scalar-identity projection:

$$G^Z_{j\ell h} = \sum_{i > j,\, g : h(g) = h} (J^Z_{ij\ell g})^\top J^Z_{ij\ell g}, \quad m^Z_{j\ell h} = \frac{\mathrm{tr}(G^Z_{j\ell h})}{d_Z}, \quad \mathcal{L}_{\mathrm{mt}} = \sum_{j,\ell,h,Z} m^Z_{j\ell h} \|\Delta Z_{j\ell h}\|_2^2, \quad Z \in \{K, V\}. \tag{8}$$

Here $d_K = d_{k,\ell}$, $d_V = d_v$, and ΔZ is the corresponding K- or V-path component of ΔC; $m^Z_{j\ell h}$ is the Frobenius-optimal scalar-identity projection. After stacking N causally matched rows, branch-aware processing produces

$$\mathbf{m}^Z_{\ell h} = (m^Z_{1\ell h}, \dots, m^Z_{N\ell h})^\top, \qquad \mathbf{d}^Z_{\ell h} = \mathrm{BlockProcess}_Z(\mathbf{m}^Z_{\ell h}), \qquad D^Z_{\ell h} = \mathrm{diag}(\mathbf{d}^Z_{\ell h}) \succ 0.$$

The resulting positive diagonal matrix $D^Z_{\ell h}$ supplies the weights for both Selection and Fit through the approximation chain $\mathcal{L}_{\mathrm{attn}} \to \mathcal{L}_{\mathrm{local}} \to \mathcal{L}_{\mathrm{mt}}$. In implementation, K-path traces are scaled by $d_{k,\ell}$; each branch-, layer-, and head-specific statistic is winsorized at a cap of 20 and shrunk with $\alpha = 0.5$ before forming $D^Z_{\ell h}$. The full pullback and processing derivations appear in section A.3 and eq. (A.17).

3.3 Attention-Guided Selection and Closed-Form Fit

The target-local weights guide two coupled steps: target-weighted CKA selects source layers, and a head-specific weighted fit maps their representations into each target KV head. After DePos and causal alignment, target-weighted CKA ranks source layers using projected K/V views and target KV-head weights (Kornblith et al., 2019):

$$\bar{D}^Z_\ell = \frac{1}{H_{\mathrm{KV},T}} \sum_h D^Z_{\ell h}, \quad \bar{D}^Z_{\ell,\mathrm{sel}} = (\bar{D}^Z_\ell)_{1:N_{\mathrm{sel}},1:N_{\mathrm{sel}}}, \quad \gamma_{\ell\iota} = \frac{1}{2} \sum_{Z\in\{K,V\}} \mathrm{CKA}_{\bar{D}^Z_{\ell,\mathrm{sel}}}(U^Z_\iota, Y^Z_\ell), \quad \mathcal{I}_\ell = \mathrm{TopK}_\iota(\gamma_{\ell\iota}, k). \tag{9}$$

Here $U^Z_\iota, Y^Z_\ell \in \mathbb{R}^{N_{\mathrm{sel}}\times d_{\mathrm{proj}}}$ use $N_{\mathrm{sel}} = \min(N, 2048)$ rows; $H_{\mathrm{KV},T}$ and k set the target-head count and layer budget. K and V share $\mathcal{I}_\ell$; Fit uses full features and branch-specific metrics.

Given a target layer ℓ, KV head h, and branch Z, the selected support $\mathcal{I}_\ell$ and eq. (4) supply $F = F^Z_\ell$. Let $Y = Y^Z_{\ell h}$ denote the target response and $D = D^Z_{\ell h}$; their weighted centers are

$$\mu_F = \frac{\mathbf{1}^\top D F}{\mathbf{1}^\top D \mathbf{1}}, \qquad \mu_Y = \frac{\mathbf{1}^\top D Y}{\mathbf{1}^\top D \mathbf{1}}, \qquad F_c = F - \mathbf{1}\mu_F, \qquad Y_c = Y - \mathbf{1}\mu_Y.$$

With an unregularized bias and $\lambda_{\mathrm{eff}} > 0$, the head-specific affine map is obtained from the weighted ridge problem (Hoerl & Kennard, 1970):

$$\begin{aligned} (W^*, c^*) &= \arg\min_{W,c} \left\| D^{1/2}(FW + \mathbf{1}c - Y) \right\|_F^2 + \lambda_{\mathrm{eff}} \|W\|_F^2, \\ W^* &= (F_c^\top D F_c + \lambda_{\mathrm{eff}} I_{d_F})^{-1} F_c^\top D Y_c, \qquad\qquad c^* = \mu_Y - \mu_F W^*. \end{aligned} \tag{10}$$

TACIT solves each (ℓ, h, Z) independently, and implements the inverse as a stable linear solve. We use the branch-adaptive scale $\lambda_{\mathrm{eff}} = \rho \max\{\mathrm{tr}(F_c^\top D F_c)/d_F, \varepsilon\}$ with default $\rho = 10^{-4}$ and $\varepsilon = 10^{-12}$. Ridge solver details appear in section A.4.3.

4 Experiments

4.1 Setup

Models. We evaluate ten models spanning 560M–32B parameters across five model families: Qwen3-1.7B/4B/8B/14B/32B (Yang et al., 2025), Llama-3.2-3B and Llama-3.1-8B (Dubey et al., 2024), Mistral-7B (Jiang et al., 2023), SmolLM3-3B (Bakouch et al., 2025), and BLOOM-560M (Le Scao et al., 2022). The pairs span heterogeneous tokenizers, depths, KV-head layouts, and positional mechanisms. Each direction uses one frozen mapper with $k = 2$ source layers per target layer. Detailed mapper specifications appear in Appendix C.2.2.

Benchmarks. Our evaluation covers three settings: multiple-choice reasoning, open-vocabulary continuation, and answer generation. For multiple-choice reasoning, we report accuracy on six benchmarks: HellaSwag (HS) (Zellers et al., 2019), CICERO (CIC) (Ghosal et al., 2022), ToMATO (TOM) (Shinoda et al., 2025), SWAG (SW) (Zellers et al., 2018), PIQA (PI) (Bisk et al., 2020), and Story Cloze (SC) (Mostafazadeh et al., 2016). We report continuation perplexity (PPL) on WikiText-103 (Merity et al., 2016). For answer generation, we report exact match (EM) and F1 on SQuAD v1.1 (Rajpurkar et al., 2016).

Baselines. We compare four learned KV-cache translators: C2C's neural projection and gated fusion (Fu et al., 2026), MoT's translator mixture and context correction (Lee et al., 2026), LCF's compact joint K/V translation and compression (Rossi et al., 2026), and LSC's shared latent space with model-specific adapters (Dery et al., 2026). Together, they span gated projection, translator mixtures, compact compression, and shared latent adaptation, providing a broad reference for transfer quality and efficiency. All methods use the same 96 FineWiki (Penedo, 2025) documents per direction for training or construction, with source and target models frozen, and share the per-item replay boundary and task-specific protocols (Sections B.2 and B.3); the learned translators are thus trained per direction on exactly the data from which TACIT computes its closed-form statistics.

4.2 Main Results

Cross-family transfer. Cross-family transfer requires translated memory to support the target model's downstream predictions. The HellaSwag results in Table 1 show that TACIT achieves this across five model families with different tokenizers and KV-head layouts. Across all 13 cross-family directions, TACIT retains a median 78.70% of target-native HellaSwag accuracy, with 11 directions

exceeding 75% retention. The prefix-scaling study on Qwen3-8B→Llama-3.1-8B retains 92.4–95.4% of target-native next-token accuracy while reducing end-to-end TTFT from 32.8–1762.3 ms to 21.6–168.2 ms, yielding 1.52×–10.48× speedups from 1K to 32K prefixes (table 2).

Table 1: Cross-family accuracy and target-relative retention on HellaSwag (13 directions).

Source → Target	Accuracy (%) ↑		Ret. (%)
	Target Prefill	TACIT	
Qwen3-8B → Llama-3.1-8B	77.00	61.96	80.47
Mistral-7B → Llama-3.2-3B	68.62	52.80	76.94
BLOOM-560M → Mistral-7B	81.17	43.86	54.03
Mistral-7B → Qwen3-8B	71.99	56.65	78.70
SmolLM3-3B → Mistral-7B	81.17	63.16	77.82
Llama-3.2-3B → Qwen3-14B	76.73	62.05	80.87
Qwen3-4B → Llama-3.2-3B	68.62	54.38	79.25
Mistral-7B → Qwen3-14B	76.73	60.79	79.23
Llama-3.1-8B → Qwen3-32B	79.85	67.64	84.70
Qwen3-1.7B → Llama-3.2-3B	68.62	53.74	78.32
Qwen3-1.7B → Mistral-7B	81.17	63.53	78.27
BLOOM-560M → Llama-3.2-3B	68.62	47.00	68.50
Llama-3.2-3B → BLOOM-560M	33.76	31.15	92.27

Ret. = $100 \times \mathrm{Acc}_{\mathrm{TACIT}} / \mathrm{Acc}_{\mathrm{Target\ Prefill}}$.

Table 2: Continuation quality and end-to-end TTFT as prefix context grows.

Continuation quality (%) ↑			
Prefix	Target native	TACIT	Ret.
1K	60.6	**57.8**	**95.4**
4K	60.6	**56.0**	**92.4**
16K	62.2	**59.3**	**95.3**
32K	72.3	**68.6**	**95.0**
End-to-end TTFT (ms) ↓			
Prefix	Target prefill	TACIT TTFT	Speedup
1K	32.8	**21.6**	**1.52×**
4K	131.9	**35.7**	**3.70×**
16K	672.8	**91.3**	**7.37×**
32K	1762.3	**168.2**	**10.48×**

Multitask performance. TACIT leads all six multiple-choice benchmarks in accuracy and handoff latency (table 3). With one frozen mapper per direction across tasks, it reaches 55.35% mean accuracy (95% CI: [54.83, 55.87]) and 6.13 ms mean handoff latency. Relative to LCF, TACIT's paired accuracy gain of 0.85 pp (95% CI: [0.65, 1.06]) accompanies a 72.0% latency reduction. Overall handoff is 3.58×–11.64× faster than C2C, MoT, LCF, and LSC (6.77× on average). Beyond multiple-choice tasks, we evaluate TACIT on text continuation and answer generation (table 4). On SQuAD, TACIT reaches 17.48 EM/31.29 F1, improving on LCF by 3.74/3.21 points with a 3.13× faster handoff at 8.33 ms. On WikiText-103, it achieves 15.15 PPL and the lowest mean handoff latency of 22.07 ms. Across these workloads, TACIT preserves target-relative quality while delivering the lowest measured cache-handoff latency.

Table 3: Multitask transfer accuracy and cache-handoff latency. Results average the six directions within each task.

Method	Metric	Benchmarks						Overall	Ret. (%)
		HS	CIC	TOM	SW	PI	SC		
C2C	Acc. (%) ↑	47.33	25.39	37.44	56.52	69.70	54.00	48.40	72.87
	Handoff (ms) ↓	35.82	40.08	87.67	35.73	36.35	36.24	45.31	
MoT	Acc. (%) ↑	55.44	32.09	42.39	62.70	67.44	59.82	53.31	80.26
	Handoff (ms) ↓	17.91	23.33	62.66	20.95	22.41	17.06	27.38	
LCF	Acc. (%) ↑	56.86	32.97	45.52	63.22	68.69	59.71	54.49	82.04
	Handoff (ms) ↓	17.12	19.63	42.85	17.22	17.27	17.37	21.91	
LSC	Acc. (%) ↑	41.78	23.64	34.98	46.04	60.11	53.90	43.41	65.35
	Handoff (ms) ↓	73.94	78.50	80.43	64.91	64.46	65.58	71.30	
TACIT	Acc. (%) ↑	**57.02**	**33.64**	**47.12**	**63.68**	**70.71**	**59.92**	**55.35**	**83.33**
	Handoff (ms) ↓	**4.25**	**5.80**	**17.45**	**2.67**	**2.21**	**4.39**	**6.13**	

Training-free mapper construction. TACIT constructs affine mappers directly through target-attention-guided Selection and closed-form Fit, avoiding the backpropagation and iterative optimization used to train nonlinear translators. Paired unlabeled prefills provide the calibration states and attention statistics for construction. On a single H800, these stages construct a mapper in 2.21 minutes, reducing LCF's 3.12-minute cost by 29.2% (table C.3). Including shared KV extraction,

the complete path takes 5.16 versus 6.07 minutes. Method-stage mean peak memory is 29.47 GiB, 13.1% below LCF (fig. 3). Taken together with the transfer results above, these measurements show that TACIT reduces mapper-construction cost without sacrificing downstream task quality.

Table 4: Continuation and answer generation with mean cache-handoff latency.

Method	WikiText-103		SQuAD v1.1		
	PPL ↓	Handoff (ms) ↓	EM (%) ↑	F1 (%) ↑	Handoff (ms) ↓
C2C	119.49	100.62	4.70	12.12	48.00
MoT	17.04	77.69	13.62	27.80	29.54
LCF	**14.18**	52.07	13.74	28.08	26.08
LSC	22.10	28.41	11.02	22.89	9.11
TACIT	15.15	**22.07**	**17.48**	**31.29**	**8.33**

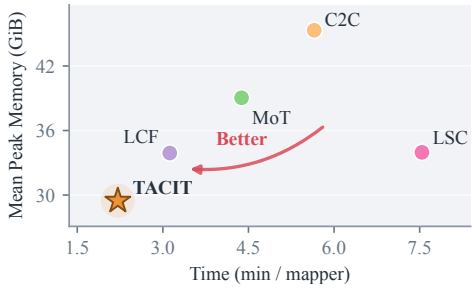

Figure 3: Mapper construction cost over TACIT and baselines.

Mapper capacity and cost. The source-layer budget k sets the number of source layers per target layer. On Qwen3-8B→Llama-3.1-8B and Mistral-7B→Qwen3-8B, $k = 2$ achieves 60.55% mean HellaSwag accuracy, versus 57.81% for $k = 1$ and 60.16% for $k = 4$. Compared with $k = 4$, it nearly halves parameters (142.68M versus 285.28M) and reduces mean resident-cache handoff p50 from 6.39 to 4.04 ms (table 5). This suggests that useful, complementary information is concentrated in a few selected source layers. Selection identifies this information, while Fit preserves its effect on target attention. The mapper stores one $d_F \times d_Z$ affine map per target layer, KV head, and branch, independent of prefix length, and handoff applies these maps as one batched matrix product per target layer followed by RePos, without any target forward pass.

Table 5: Mapper capacity, accuracy, and resident-cache handoff cost, averaged across two directions. Experimental protocol: Appendix C.2.3.

Source layers	$k = 1$	$k = 2$ **(default)**	$k = 4$
Accuracy (%)	57.81	**60.55**	60.16
95% CI (%)	[53.81, 61.82]	[56.64, 64.45]	[56.15, 64.06]
Parameters (M)	71.37	142.68	285.28
Handoff p50 / p90 (ms)	2.86 / 3.45	4.04 / 4.42	6.39 / 6.52

4.3 Ablation Studies

Selection and Fit. Selection chooses source layers, while Fit reconstructs their K/V features using target-attention weights. We cross coordinate and attention-guided variants of both at fixed affine capacity (table 6). Against fully coordinate construction, Fit alone gains 5.86–20.90 points across four directions; Selection alone gains 16.21/15.04 points on the Qwen-source pairs. Full TACIT gains 5.47–21.88 points, with every paired 95% interval above zero. Thus, attention-weighted Fit raises accuracy across all four directions with source layers unchanged, while attention-guided Selection alone adds 15.04–16.21 points on the two Qwen-source pairs with coordinate Fit unchanged (sections C.3.2 and C.5).

Target-local attention metric. Cache errors affect the target through attention routing and value readout, so TACIT's metric must capture their impact on attention outputs. We compare the full pullback geometry $J_T^\top J_T$ with nonlinear attention-output distortion under K/V perturbations. At $\epsilon = 0.25$, it achieves Spearman $\rho = 0.9900$–0.9994 across all nine direction×branch comparisons (fig. 4), showing that it ranks cache errors by their local impact on target computation. Across perturbation scales, V correlation stays near 1 because value readout is linear, while K and joint vary with nonlinear softmax routing. These responses support TACIT's separate K/V metrics.
The mean-trace metric used in TACIT's closed-form Fit correlates more strongly with attention-output distortion than coordinate distance across all nine direction×branch comparisons (fig. 5 and table C.8): median Spearman correlation rises from 0.368 to 0.754 for K, while V/joint reach

Table 6: HellaSwag accuracy (%) with bootstrap 95% CIs. Bars: 0–100%; bold: row maximum.

Transfer direction	Coordinate	Selection only	Fit only	**Full TACIT**
Qwen3-8B → Llama-3.1-8B	45.90 [41.60, 50.20]	62.11 [57.81, 66.22]	52.73 [48.44, 57.03]	**62.89** [58.59, 66.99]
Mistral-7B → Llama-3.2-3B	33.40 [29.30, 37.50]	31.45 [27.54, 35.55]	54.30 [50.00, 58.59]	**55.27** [50.98, 59.57]
Qwen3-4B → Llama-3.2-3B	38.28 [33.98, 42.38]	53.32 [49.02, 57.62]	56.25 [51.95, 60.55]	**56.64** [52.34, 60.94]
Mistral-7B → Qwen3-8B	52.73 [48.44, 57.03]	50.98 [46.68, 55.27]	**58.59** [54.30, 62.89]	58.20 [53.91, 62.50]

0.997/0.979. With Selection fixed, mean-trace Fit also reduces geometric-mean attention-output distortion in all 18 comparisons with coordinate Fit (table C.9). Mean-trace weighting thus better ranks cache errors by their impact on target attention.

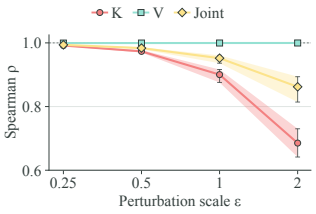

Figure 4: Full local pullback preserves attention-distortion ordering (95% CIs).

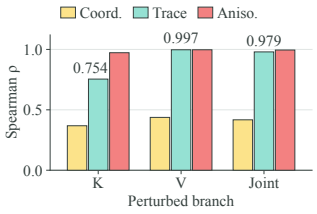

Figure 5: Mean-trace pullback fidelity across three directions.

Table 7: HellaSwag accuracy (%) for mapper classes (Qwen3-1.7B→Llama-3.2-3B).

Mapper	Coordinate	Attention-guided
Full affine	25.83	53.74
Factorized linear	27.79	51.38
GELU MLP	36.51	50.35

Mapper function class. We compare full affine, factorized linear, and GELU mappers at matched parameter counts under coordinate and attention-guided construction (table 7). On the full 10,042-item validation split, coordinate construction reaches 25.83%, 27.79%, and 36.51% for full affine, factorized linear, and GELU, respectively. Attention-guided construction reaches 53.74%, 51.38%, and 50.35%, respectively, with full affine remaining the strongest mapper in that geometry. These results support TACIT's attention-guided affine design, which preserves target computation through closed-form construction.

4.4 Mechanism Analysis

TACIT's explicit affine K/V maps let us trace how transferred caches affect target attention and predictions. We combine layerwise attention diagnostics and accuracy measurements across six transfer directions with interventions on branches, layers, and sensitivity weights (section C.4).

Interpreting the mapped values. At each layer and head, TACIT reconstructs values as $\widehat{V} = F^V W^V + \mathbf{1}c^V$. With native Q/K and suffix V fixed, the output error is $\Delta O = A_{\text{native,prefix}} \Delta V_{\text{prefix}}$, so each row's contribution is scaled by target attention. The decomposition therefore ranks cache errors by readout impact rather than coordinate distance. Figure 6 compares native and mapped V with their attention-weighted outputs; their alignment indicates preserved target readout rather than merely similar values. On Qwen3-1.7B→Llama-3.2-3B, TACIT reduces V-only output error by 6.95% relative to GELU across four held-out passages.

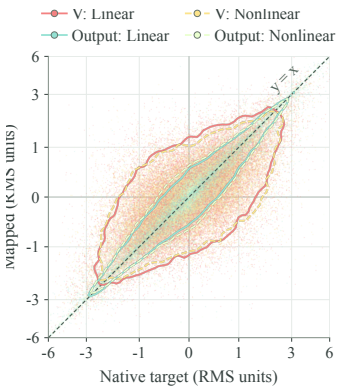

Figure 6: Native and mapped V and V-only outputs.

Tracing routing and readout. We trace K-routing with attention KL and V-readout error with native Q/K fixed (fig. 7). Selection reduces V-readout error by 84–91% at layers 3–5 on Qwen3-1.7B→Llama-3.2-3B and by 87.8% at layer 4 on Qwen3-8B→Llama-3.1-8B. With Selection fixed, Fit reduces layer-2 attention KL/V-readout error by 60.4%/38.2% and 78.5%/43.8% in the two di-

rections, respectively. The profiles locate Selection's readout gains in early layers and show how Fit refines routing and readout on fixed source features. With Selection fixed, Fit reduces joint attention-output error and raises accuracy in all six directions reported in table C.14.

Interpreting sensitivity at cache positions. Fit weights encode the sensitivity of target attention to individual cache positions. We test this correspondence by shuffling weights while preserving their distribution and keeping Selection fixed. On Qwen3-1.7B→Llama-3.2-3B, true weights raise accuracy from 53.52% to 55.86% and reduce attention KL from 0.159 to 0.112. Shuffling lowers accuracy by 1.95 points and returns KL to 0.162 (table C.13). With the distribution unchanged, this drop links Fit's gains to correctly locating sensitivity in the cache. The parallel changes in KL and accuracy indicate that Fit improves transfer by aligning weight mass with positions that matter to the target computation, rather than simply rescaling the cache as a whole.

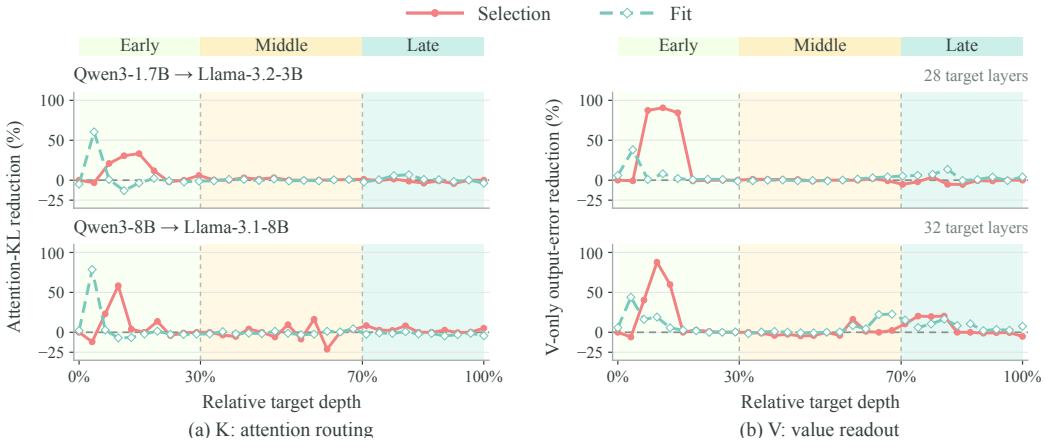

Figure 7: Layerwise reductions in attention KL and V-only output error for two transfer directions. Selection is measured against Coordinate; Fit against uniform fitting under fixed TACIT Selection.

Linking internal effects to answer accuracy. To connect the attention diagnostics to model behavior, we apply Selection to the explicit K/V maps in individual branches or layer blocks, retaining coordinate Fit. On Qwen3-1.7B→Llama-3.2-3B, V-only Selection gains 27.15 points and early-layer K+V Selection gains 27.34 points over the all-coordinate baseline. The corresponding gains on Qwen3-8B→Llama-3.1-8B are 16.60/16.80 points (tables C.12 and C.14). These interventions link the localized readout improvements to answer accuracy, connecting TACIT's explicit affine maps, their effects on target computation, and model behavior. Together, the profiles and interventions assign the two branches distinct roles: keys decide where the target attends and values decide what it reads out, so the same cache error can be costly for one branch and negligible for the other. This division is why TACIT retains separate K- and V-path metrics instead of a single cache-distance objective. Because every effect is read off an explicit affine map, it can be localized to a branch, a layer block, or a set of cache positions, which is what allows the diagnostics above to attribute accuracy to specific parts of the transferred cache.

5 Conclusion

TACIT establishes a training-free, closed-form route to KV-cache transfer across heterogeneous language models: target attention sensitivity selects which source layers to read and weights each head's K/V fit, solved in closed form from paired unlabeled prefills into one frozen mapper per direction. Across five model families and 13 cross-family directions, it retains 83.33% of target-native accuracy with 6.13 ms mean handoff, and reaches $1.52\times$–$10.48\times$ TTFT speedups on 1K–32K prefixes while retaining 92.4–95.4% of next-token accuracy. Because each mapper is a set of explicit affine maps fitted once from paired prefills, the same procedure applies to any pair of frozen models without modifying either. Selection preserves how the target routes keys and weighted fitting preserves how it reads values, turning model-private KV state into a reusable interface measured by the target computation it preserves.

AI USE STATEMENT

We did not use generative AI tools for tasks requiring disclosure, including research ideation, methodological or experimental design, implementation, or interpretation of results. Generative AI tools were used solely to edit author-written manuscript text for grammar, clarity, and concision. All AI-assisted edits were reviewed and revised by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

This work involves no human subjects or new personal data. We use publicly released models and benchmark datasets under their licenses for research purposes.

REPRODUCIBILITY STATEMENT

Algorithmic details, experimental protocols, and additional results appear in Appendices A, B, and C; implementation scripts, environment specifications, model manifests, and item-level results are provided in the supplementary material.

REFERENCES

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.298.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, Kashif Rasul, Nathan Habib, Clémentine Fourrier, Hynek Kydlicek, Guilherme Penedo, Hugo Larcher, Mathieu Morlon, Vaibhav Srivastav, Joshua Lochner, Xuan-Son Nguyen, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM3: smol, multilingual, long-context reasoner. Hugging Face Blog, July 2025. URL https://huggingface.co/blog/smollm3.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. In *Advances in Neural Information Processing Systems*, volume 34, pp. 225–236, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/01ded4259d101feb739b06c399e9cd9c-Abstract.html.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239.

Lucio M. Dery, Zohar Yahav, Henry Prior, Qixuan Feng, Jiajun Shen, and Arthur Szlam. Latent space communication via K-V cache alignment. *arXiv preprint arXiv:2601.06123*, 2026. URL https://arxiv.org/abs/2601.06123.

Abhimanyu Dubey et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783v1.

Tianyu Fu, Zihan Min, Hanling Zhang, Jichao Yan, Guohao Dai, Wanli Ouyang, and Yu Wang. Cache-to-cache: Direct semantic communication between large language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 43130–43158, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/474ada926b331d78f06d95e8913111cc-Abstract-Conference.html.

Deepanway Ghosal, Siqi Shen, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. CICERO: A dataset for contextualized commonsense inference in dialogues. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5010–5028, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.344.

In Gim, Guojun Chen, Seung-seob Lee, Nikhil Sarda, Anurag Khandelwal, and Lin Zhong. Prompt cache: Modular attention reuse for low-latency inference. In *Proceedings of Machine Learning and Systems*, volume 6, pp. 325–338, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/a66caa1703fe34705a4368c3014c1966-Abstract-Conference.html.

Taekyung Heo, Rasoul Shafipour, Ritchie Zhao, Maximilian Golub, Mohammad Mahdi Kamani, Ritika Borkar, Makesh Tarun Chandran, Pantea Zardoshti, and Bita Darvish Rouhani. Cross-model KV cache transfer in LLM families: A closed-form linear mapping for prefill reuse. *arXiv preprint arXiv:2608.03893*, 2026. URL https://arxiv.org/abs/2608.03893.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023. URL https://arxiv.org/abs/2310.06825.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, 2023. URL https://arxiv.org/abs/2309.06180.

Teven Le Scao et al. BLOOM: A 176B-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. URL https://arxiv.org/abs/2211.05100.

Jin-woo Lee, Minkyung Song, Junghyun Oh, Seunghoon Han, Soyoung Park, Gwangseon Jang, and Sungsu Lim. Mixture-of-translators: Translating KV caches across heterogeneous large language models. *arXiv preprint arXiv:2607.28979*, 2026. URL https://arxiv.org/abs/2607.28979.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020. URL https://jmlr.org/papers/v21/17-678.html.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016. URL https://arxiv.org/abs/1609.07843.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper understanding of commonsense stories. In Kevin Knight, Ani Nenkova, and Owen Rambow (eds.), *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 839–849, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1098.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8sSqNntaMr.

Guilherme Penedo. FineWiki. Hugging Face Datasets, 2025. URL https://huggingface.co/datasets/HuggingFaceFW/finewiki.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2108.12409.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392. Association for Computational Linguistics, 2016. doi: 10.18653/v1/D16-1264.

Maximillian Rossi, Prajwal Raghunath, and Eugene Wu. Latent cache flow: Model-to-model communication without text. In *AdaptFM Workshop at the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=PvnR2LDCOs.

Kazutoshi Shinoda, Nobukatsu Hojo, Kyosuke Nishida, Saki Mizuno, Keita Suzuki, Ryo Masumura, Hiroaki Sugiyama, and Kuniko Saito. ToMATO: Verbalizing the mental states of role-playing LLMs for benchmarking theory of mind. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(2):1520–1528, 2025. doi: 10.1609/aaai.v39i2.32143.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021. URL https://arxiv.org/abs/2104.09864.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. Mixture-of-agents enhances large language model capabilities. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=h0ZfDIrj7T.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Ahmed Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=BAakY1hNKS.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.17453.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 93–104, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1009.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H_2O: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.14048.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2000.

A Mathematical Foundations and Algorithms

All equations use one-based mathematical indices unless a serialized mapper index is marked with a tilde.

A.1 Task, Interface, and Tensor Shapes

We use the transfer contract in eq. (1) and the local-attention objective in eq. (5). For replay length R, the transported span has $n_c = n - R$ tokens, $t_{1:n_c}$, and the target replays $r = t_{n_c+1:n}$. The following conventions specify the tensors used to implement these operators.

Let $b \in \{K, V\}$ identify the cache branch (the same branch index denoted by Z in the main text); B the batch size; L_S, L_T the layer counts; $H_{\mathrm{KV},S}, H_{\mathrm{KV},T}$ the KV-head counts; and $d_{b,S}, d_{b,T}$ the per-head widths. Query position is i, cache slots are j, u, source layer and head are ι, h_S, and target layer and head are ℓ, h; target query head g is served by KV head $h(g)$. The logical cache stacks have shapes

$$\begin{aligned} C_S^b &\in \mathbb{R}^{B\times n_S\times L_S\times H_{\mathrm{KV},S}\times d_{b,S}}, \\ C_T^b, \widehat{C}_T^b &\in \mathbb{R}^{B\times n_c\times L_T\times H_{\mathrm{KV},T}\times d_{b,T}}. \end{aligned} \tag{A.1}$$

These operator shapes are independent of the serialized memory order. The current runtime serializes each target layer as $[B, H_{\mathrm{KV},T}, n_c, d_{b,T}]$ and may transpose from a time-major $[B, n_c, H_{\mathrm{KV},T}, d_{b,T}]$ view.

Stacking causally aligned calibration slots gives N paired observations, raw mean-trace statistics $\mathbf{m}^b \in \mathbb{R}^{N\times L_T\times H_{\mathrm{KV},T}}$, and an ordered selection $\mathcal{I} \in \mathbb{N}^{L_T\times k}$. Once the k source layers are assembled, the relevant dimensions are

$$\begin{aligned} d_F^b &= kH_{\mathrm{KV},S}d_{b,S}, \quad F_\ell^b \in \mathbb{R}^{N\times d_F^b}, \quad Y_{\ell h}^b \in \mathbb{R}^{N\times d_{b,T}}, \\ W_{\ell h}^b &\in \mathbb{R}^{d_F^b\times d_{b,T}}, \quad c_{\ell h}^b \in \mathbb{R}^{d_{b,T}}. \end{aligned} \tag{A.2}$$

The implementation alias is $p_b = d_F^b$. Independent head-wise solutions may be packed as $W_\ell^b = [W_{\ell 1}^b \mid \cdots \mid W_{\ell H_{\mathrm{KV},T}}^b]$ with a matching concatenated bias c_ℓ^b. The mathematical operator retains a batch axis; the current cross-tokenizer implementation instantiates the operator with $B = 1$.

A.2 Causal Alignment and Positional Canonicalization

The main text writes the causal gather as $A_{S\to T}(x)$, denotes its positional transforms by DePos and RePos, and uses Cat_ℓ for ordered layer/head feature assembly; this section gives their branch-specific definitions and execution order.

Let e_j^S and e_u^T be character-end offsets in the same canonical text coordinate system. The dense relation $M_{uj}(x) = \mathbf{1}[e_j^S \le e_u^T]$ describes causal availability; TACIT realizes it as a one-hot operator that gathers the last fully visible source token:

$$\begin{aligned} \pi_x(u) &= \max\big(\{j : e_j^S \le e_u^T\} \cup \{0\}\big), \qquad [A_{S\to T}(x)]_{uj} = \mathbf{1}[\pi_x(u) > 0 \wedge j = \pi_x(u)], \\ [A_{S\to T}(x)Z]_u &= \begin{cases} Z_{\pi_x(u)}, & \pi_x(u) > 0 \\ 0, & \pi_x(u) = 0 \end{cases}. \end{aligned} \tag{A.3}$$

Each target slot gathers one causally visible source token; a coarse source token may serve several target slots. Equal endpoints select the greatest source index. An empty-history row remains zero across every source layer, head, and channel; the later affine bias may still produce an output for that row.

Endpoints come from fast-tokenizer character-offset mappings with `add_special_tokens =False`. The target tokenizer supplies target IDs and character endpoints for alignment. Missing endpoints are computed from the shared text with the corresponding tokenizer; callers provide both endpoint sequences when the shared text is unavailable. Special tokens must be included explicitly in both span and position conventions when required. Calibration truncates text at a target-token boundary and retains spans whose token counts and boundaries are stable under retokenization. Calibration uses nonempty sequences; inference-time handoff requires $n \ge 1$ and $0 \le R < n$, retaining at least one transported token.

Alignment is applied per sample using the unpadded lengths and a request-specific matrix. The calibration operator $A_{\mathrm{cal}}(x)$ extends the same row rule to all target observations used for statistics, while inference-time $A_{S\to T}(x)$ has exactly n_c rows.

For model $M \in \{S, T\}$, let p_j^M be token j's native cache position and $R_{M,a}(p_j^M)$ the cache-side positional action at layer a. DePos and RePos act row-wise as

$$\begin{aligned} \mathcal{D}^K_{M,a}(Z)_{j:} &= Z_{j:} R_{M,a}(p_j^M)^{-1}, \quad \mathcal{D}^V_{M,a}(Z) = Z, \\ \mathcal{R}^K_{M,a}(U)_{j:} &= U_{j:} R_{M,a}(p_j^M), \qquad \mathcal{R}^V_{M,a}(U) = U. \end{aligned} \tag{A.4}$$

Source caches are then de-positioned and gathered as $\widetilde{C}^b_{S,\iota,h_S} = A_{S\to T}(x)\mathcal{D}^b_{S,\iota}(C^b_{S,\iota,h_S})$; paired target caches use the corresponding target DePos operator during construction. Canonicalization removes the explicit cache-side positional action while preserving positional information already encoded in the hidden states.

For standard RoPE, the source key is de-rotated with its own layer-specific coefficients and the predicted target key is rotated with the target layer's coefficients at the materialized position. Values use the identity action. For ALiBi and NoPE layers, $R_{M,a}(p) = I$; target-native ALiBi bias remains in the attention logits. TACIT supports reversible cache-side positional actions, including RoPE, ALiBi, and NoPE. Dynamic-RoPE coefficients come directly from each model under its native position IDs, sequence length, and configuration.

Let $\mathcal{H}_S$ be the source KV-head index set. The complete branch-wise affine K/V transport is

$$\boxed{\begin{aligned} F^b_\ell(x) = \mathrm{Cat}_\ell(\widetilde{C}^b_S) &:= \Big\|_{\iota\in\mathcal{I}_\ell,\, h_S\in\mathcal{H}_S} A_{S\to T}(x)\mathcal{D}^b_{S,\iota}\big(C^b_{S,\iota,h_S}(x)\big)\,, \\ \widehat{C}^b_{T,\ell,h}(t_{1:n_c}) &= \mathcal{R}^b_{T,\ell}\big[F^b_\ell(x)W^b_{\ell h} + \mathbf{1}c^b_{\ell h}\big]\,, \qquad b \in \{K, V\} \end{aligned}}. \tag{A.5}$$

The concatenation is Feature Assembly: it traverses the ordered selected layers and, within each layer, source heads in increasing order before the rectangular affine map is applied. The prediction is then packed into the target-native layout during cache materialization. Assembly before the map is essential: the construction accommodates different head counts and widths across families with different head counts and widths while preserving native head indexing.

These operations are order-sensitive. Aligning before source De-Pos couples tokenizer boundaries to the source positional phase; fitting after target Re-Pos couples the content map to a particular target slot. TACIT therefore executes text alignment, positional coordinates, and content translation as separate, ordered operators.

A.3 Target-Attention Pullback

The metric is constructed on complete native calibration prefills $C_{T,\mathrm{cal}} = \mathrm{Prefill}_T(\mathrm{Tok}_T(x_{\mathrm{cal}}))$, distinct from the $t_{1:n_c}$ handoff reference cache in section A.1. All relations $i > j$ below are evaluated within one such calibration span before observations are stacked.

A.3.1 Exact local attention Jacobians

At target layer ℓ, query head g is served by KV head $h(g)$. At the target-native operating point,

$$\begin{aligned} z_{ij\ell g} &= \tau_\ell q_{i\ell g}^\top k_{j\ell h(g)} + \beta_{ij\ell g}, \\ a_{ij\ell g} &= \mathrm{softmax}_j(z_{i\cdot\ell g}), \\ o_{i\ell g} &= \sum_u a_{iu\ell g} v_{u\ell h(g)}. \end{aligned} \tag{A.6}$$

Here, $\tau_\ell = d_{k,\ell}^{-1/2}$ is the target layer's native attention scale and $\beta_{ij\ell g}$ contains the native causal mask and positional bias. Construction-time target prefills supply these quantities; inference-time handoff uses the resulting mapper.

Values enter the attention output linearly. Combining $\partial a_{iu}/\partial z_{ij} = a_{iu}(\mathbf{1}[u = j] - a_{ij})$ with the attention equations gives both branch Jacobians

$$\begin{aligned} J^V_{ij\ell g} &= \frac{\partial o_{i\ell g}}{\partial v_{j\ell h(g)}} = a_{ij\ell g} I, \\ J^K_{ij\ell g} &= \frac{\partial o_{i\ell g}}{\partial k_{j\ell h(g)}} = \tau_\ell a_{ij\ell g}(v_{j\ell h(g)} - o_{i\ell g}) q_{i\ell g}^\top. \end{aligned} \tag{A.7}$$

Key sensitivity combines attention mass with the query direction and value residual.

A.3.2 Full local Gram and same-slot blocks

For cache errors e_j^K, e_j^V, the first-order response of a fixed consumer is

$$\Delta o_{i\ell g} \approx \sum_j \left(J^K_{ij\ell g} e^K_{j\ell h(g)} + J^V_{ij\ell g} e^V_{j\ell h(g)} \right). \tag{A.8}$$

Stacking all slots and branches into e yields a positive-semidefinite local Gram, whose blockwise expansion is

$$\begin{aligned} \mathcal{L}_{\text{local}} &:= e^\top \left(\sum_{i,\ell,g} J_{i\ell g}^\top J_{i\ell g} \right) e \approx \mathcal{L}_{\text{attn}}, \\ \|\Delta o_{i\ell g}\|_2^2 &\approx \sum_{b,c\in\{K,V\}} \sum_{j,u} (e_j^b)^\top (J^b_{ij\ell g})^\top J^c_{iu\ell g} e_u^c. \end{aligned} \tag{A.9}$$

The full positive-semidefinite Gram contains cross-slot blocks ($j \neq u$), same- and cross-slot K/V coupling ($b \neq c$), and channel anisotropy in the key blocks. Retaining the full Gram couples slots and branches in a prohibitively large linear system. TACIT projects this geometry into same-slot, same-path blocks, yielding separable closed-form solves that scale independently across target layers, heads, and K/V paths.

At fixed target-native queries, TACIT accumulates the same-slot blocks over subsequent prefill consumers. For slot j, target layer ℓ, and target KV head h, these consumers satisfy $i > j$ and $h(g) = h$:

$$\begin{aligned} G^V_{j\ell h} &= \sum_{i>j} \sum_{g:h(g)=h} (J^V_{ij\ell g})^\top J^V_{ij\ell g} = \left(\sum_{i>j} \sum_{g:h(g)=h} a^2_{ij\ell g} \right) I_{d_v}, \\ G^K_{j\ell h} &= \sum_{i>j} \sum_{g:h(g)=h} \tau_\ell^2 a^2_{ij\ell g} \|v_{j\ell h} - o_{i\ell g}\|_2^2 q_{i\ell g} q_{i\ell g}^\top. \end{aligned} \tag{A.10}$$

These expressions evaluate the same-slot blocks of the local first-order operator exactly. The final calibration slot has zero raw weight because it has no subsequent consumer. At inference, replay and task-suffix tokens can attend to the last materialized slot t_{n_c}.

A.3.3 Mean-trace projection

TACIT makes the local system separable by projecting each block onto scalar identity matrices. For $G \in \mathbb{R}^{d\times d}$,

$$m^* = \arg\min_m \|G - mI_d\|_F^2 = \frac{\operatorname{tr}(G)}{d}. \tag{A.11}$$

Differentiating with respect to m gives $-2\operatorname{tr}(G) + 2dm$, with second derivative $2d > 0$. This is the unique Frobenius-optimal isotropic projection and remains nonnegative for $G \succeq 0$. Applying it to the exact blocks gives

$$\begin{aligned} m^V_{j\ell h} &= \sum_{i>j} \sum_{g:h(g)=h} a^2_{ij\ell g}, \\ m^K_{j\ell h} &= \frac{1}{d_k} \sum_{i>j} \sum_{g:h(g)=h} \tau_\ell^2 a^2_{ij\ell g} \|q_{i\ell g}\|_2^2 \|v_{j\ell h} - o_{i\ell g}\|_2^2. \end{aligned} \tag{A.12}$$

The value block is already isotropic, while the key projection preserves average sensitivity through the Frobenius-optimal scalar-identity representative. The factor $1/d_k$ belongs to the canonical mean-trace definition. The implementation stores the unnormalized trace $d_k m^K$; block-local mean normalization cancels this constant when the numerical denominator floor is inactive and otherwise uses the stored raw-trace value directly. The resulting block-isotropic approximation is

$$\mathcal{L}_{\text{mt}} := \sum_{j,\ell,h} \sum_{b\in\{K,V\}} m^b_{j\ell h} \|e^b_{j\ell h}\|_2^2. \tag{A.13}$$

Thus the approximation chain is $\mathcal{L}_{\text{attn}} \approx \mathcal{L}_{\text{local}} \to \mathcal{L}_{\text{mt}}$.

A.3.4 POSITIONAL PULLBACK

The key Jacobian is measured in the target-native post-position coordinates, whereas fitting uses de-positioned content coordinates. With row convention $u_{\text{native}} = u_{\text{content}} R_{T,\ell}(p_j^T)$, a column-form residual uses

$$P^K_{T,j\ell} = R_{T,\ell}(p_j^T)^\top, \qquad e_{\text{native}} = P^K_{T,j\ell} e_{\text{content}}. \tag{A.14}$$

The pullback preserves the corresponding quadratic energy:

$$\begin{aligned} e_{\text{native}}^\top G^{K,\text{native}}_{j\ell h} e_{\text{native}} &= e_{\text{content}}^\top G^{K,\text{content}}_{j\ell h} e_{\text{content}}, \\ G^{K,\text{content}}_{j\ell h} &= (P^K_{T,j\ell})^\top G^{K,\text{native}}_{j\ell h} P^K_{T,j\ell}. \end{aligned} \tag{A.15}$$

For standard RoPE, $P^K_{T,j\ell}(P^K_{T,j\ell})^\top = I$, so cyclic trace invariance yields

$$\operatorname{tr}(G^{K,\text{content}}_{j\ell h}) = \operatorname{tr}(G^{K,\text{native}}_{j\ell h}). \tag{A.16}$$

NoPE and ALiBi cache actions are the identity special case. Trace invariance requires orthogonality. For an invertible but nonorthogonal action, the method must first compute $G^{\text{content}} = P^\top G^{\text{native}} P$ and only then take $\operatorname{tr}(G^{\text{content}})/d$.

TACIT uses a block-diagonal, block-isotropic approximation to the local attention Gauss–Newton pullback, evaluated at fixed native queries before the output projection. The approximation yields independent weighted least-squares problems for each target layer, KV head, and branch.

A.4 METRIC PROCESSING, SELECTION, AND FIT

A.4.1 BLOCK-LOCAL ROBUSTIFICATION

For each fixed (b, ℓ, h) block, define the runtime statistic $r^b_{a\ell h} = s_{b,\ell} m^b_{a\ell h}$ with $s_{K,\ell} = d_{k,\ell}$ and $s_{V,\ell} = 1$. Thus the K path processes the unnormalized trace used by the implementation, while the V path processes its canonical scalar weight. The implementation casts r to float32 and clamps it to be nonnegative. A fixed 32-step bisection finds the largest threshold $\eta_{b\ell h}$ satisfying the normalized relative-leverage cap, after which shrinkage supplies a positive coordinate floor:

$$\begin{aligned} \overline{r}^b_{\eta,\ell h} &= \frac{1}{N}\sum_{a=1}^{N} \min(r^b_{a\ell h}, \eta), \\ \widetilde{r}^b_{a\ell h} &= \frac{\min(r^b_{a\ell h}, \eta_{b\ell h})}{\max\{\overline{r}^b_{\eta_{b\ell h},\ell h}, 10^{-12}\}}, \qquad \max_a \widetilde{r}^b_{a\ell h} \le 20, \\ d^b_{a\ell h} &= (1-\alpha) + \alpha \widetilde{r}^b_{a\ell h}, \quad \alpha = 0.5, \\ \mathbf{d}^b_{\ell h} &= \text{BlockProcess}_b(\mathbf{m}^b_{\ell h}), \qquad D^b_{\ell h} = \operatorname{diag}(\mathbf{d}^b_{\ell h}). \end{aligned} \tag{A.17}$$

Winsorization is applied independently within each branch, target layer, and KV head; the cap 20 and shrinkage coefficient $\alpha = 0.5$ define the metric-processing rule.

The processed weights satisfy $d_a \ge 0.5$. When the winsorized mean exceeds 10^{-12}, both $\widetilde{r}$ and d have mean one; when the numerical floor is active, their means may be below one. An all-zero block receives uniform weights $d_a = 0.5$ after shrinkage.

A.4.2 METRIC-CONSISTENT LAYER SELECTION

Selection averages already processed head-specific weights only after the block-local operation:

$$\overline{d}^b_{a\ell} = \frac{1}{H_{\text{KV},T}} \sum_h d^b_{a\ell h}, \qquad \overline{D}^b_\ell = \operatorname{diag}(\overline{d}^b_{1\ell}, \dots, \overline{d}^b_{N\ell}). \tag{A.18}$$

This head average is used only for Selection. Fit returns to the head-specific $D^b_{\ell h}$.

Calibration features share the target timeline:

$$\begin{aligned} \widetilde{C}^b_{S,\text{cal},\iota}(x) &= A_{\text{cal}}(x) \mathcal{D}^b_{S,\iota}(C^b_{S,\iota}(x)), \\ \overline{C}^b_{T,\text{cal},\ell}(x) &= \mathcal{D}^b_{T,\ell}(C^b_{T,\text{cal},\ell}(x)), \\ C^b_{T,\text{cal},\ell}(x) &= \text{Prefill}^b_{T,\ell}(\text{Tok}_T(x)). \end{aligned} \tag{A.19}$$

Similarity is computed after canonical alignment, pairing rows through shared text endpoints.

For flattened widths $F_{b,S} = H_{\mathrm{KV},S} d_{b,S}$ and $F_{b,T} = H_{\mathrm{KV},T} d_{b,T}$, Selection uses the first $N_{\mathrm{sel}} = \min(N, 2048)$ rows and projection width $d_{\mathrm{proj}} = \min(64, F_{b,S}, F_{b,T})$. Independent source and target Gaussian projections have entries distributed as $\mathcal{N}(0, 1/d_{\mathrm{proj}})$. The fixed projection seeds are 17 for K and 18 for V. Within a branch, distinct source and target matrices are drawn sequentially from the same fixed generator. Projection is used only for Selection; Fit uses the full unprojected features.

Let $U, V \in \mathbb{R}^{N_{\mathrm{sel}} \times d_{\mathrm{proj}}}$ be the projected features and let $\overline{D}_{\mathrm{sel}}$ be the leading $N_{\mathrm{sel}} \times N_{\mathrm{sel}}$ submetric from the same block. Using the same metric for centering and covariance,

$$\begin{aligned} \mu_U &= \frac{\mathbf{1}^\top \overline{D}_{\mathrm{sel}} U}{\mathbf{1}^\top \overline{D}_{\mathrm{sel}} \mathbf{1}}, & \mu_V &= \frac{\mathbf{1}^\top \overline{D}_{\mathrm{sel}} V}{\mathbf{1}^\top \overline{D}_{\mathrm{sel}} \mathbf{1}}, \\ U_c &= U - \mathbf{1}\mu_U, & V_c &= V - \mathbf{1}\mu_V, \end{aligned} \tag{A.20}$$

$$\mathrm{CKA}_{\overline{D}_{\mathrm{sel}}}(U, V) = \frac{\|U_c^\top \overline{D}_{\mathrm{sel}} V_c\|_F^2}{\|U_c^\top \overline{D}_{\mathrm{sel}} U_c\|_F \|V_c^\top \overline{D}_{\mathrm{sel}} V_c\|_F}.$$

Projection, normalized selection weights, and CKA are computed in float32. The weight sum and both self-Gram norms use a 10^{-12} floor. A degenerate centered feature has zero numerator and returns a zero score.

For candidate source layer ι, denote the two scores by $\gamma_{\ell\iota}^K$ and $\gamma_{\ell\iota}^V$. Default TACIT uses

$$\gamma_{\ell\iota} = \tfrac{1}{2}(\gamma_{\ell\iota}^K + \gamma_{\ell\iota}^V), \qquad \mathcal{I}_\ell = (\iota_{\ell,1}, \ldots, \iota_{\ell,k}) = \mathrm{TopK}_\iota(\gamma_{\ell\iota}, k). \tag{A.21}$$

Paper indices are one-based. A serialized frozen mapper uses target entry $\ell - 1$ and stores $\widetilde{\mathcal{I}}_{\ell-1} = (\iota_{\ell,1} - 1, \ldots, \iota_{\ell,k} - 1)$. The tuple preserves the returned order of `torch.topk`; no extra lexicographic tie-break is imposed. FEATUREASSEMBLY consumes these zero-based indices directly and traverses source heads in increasing order. Construction-time calibration and inference-time handoff use exactly the same column order. Weighted CKA top-k performs metric-consistent candidate reduction before the head-specific closed-form fit.

A.4.3 HEAD-SPECIFIC CLOSED-FORM FIT

For the selected ordered tuple,

$$F_\ell^b = \mathrm{Cat}_\ell(\widetilde{C}_S^b) := \Big\|_{\iota \in \mathcal{I}_\ell,\, h_S} \widetilde{C}_{S,\iota,h_S}^b \in \mathbb{R}^{N \times d_F^b}. \tag{A.22}$$

The K and V branches share only $\mathcal{I}_\ell$. Their design matrices, widths, targets, sample metrics, parameters, and linear systems are independent. For fixed (b, ℓ, h), abbreviate $F \in \mathbb{R}^{N \times d_F}$, $Y \in \mathbb{R}^{N \times d_{b,T}}$, and $D = \mathrm{diag}(d_1, \ldots, d_N)$. TACIT solves the bias-unregularized objective

$$\min_{W,c} \left\| D^{1/2}(FW + \mathbf{1}c - Y) \right\|_F^2 + \lambda_{\mathrm{eff}} \|W\|_F^2. \tag{A.23}$$

Weighted means and centered matrices are

$$\begin{aligned} \mu_f &= \frac{\mathbf{1}^\top D F}{\mathbf{1}^\top D \mathbf{1}}, \quad \mu_y = \frac{\mathbf{1}^\top D Y}{\mathbf{1}^\top D \mathbf{1}}, \\ F_c &= F - \mathbf{1}\mu_f, \quad Y_c = Y - \mathbf{1}\mu_y. \end{aligned} \tag{A.24}$$

The stationarity condition gives $c = \mu_y - \mu_f W$. Substituting it produces the closed-form mapper

$$W^* = (F_c^\top D F_c + \lambda_{\mathrm{eff}} I_{d_F})^{-1} F_c^\top D Y_c, \qquad c^* = \mu_y - \mu_f W^*. \tag{A.25}$$

The default branch-adaptive ridge scale is

$$\boxed{\lambda_{\mathrm{eff}} = \rho \max\left\{ \frac{\mathrm{tr}(F_c^\top D F_c)}{d_F}, \varepsilon \right\}, \qquad \rho = 10^{-4}, \quad \varepsilon = 10^{-12}}. \tag{A.26}$$

A positive ridge makes the system positive definite even for rank-deficient features. The inverse denotes a stable linear solve. The implementation builds the Gram matrix and right-hand side in float32, uses Cholesky factorization when available and a general linear solve otherwise, and accepts solutions whose normalized linear-system residual is at most 5×10^{-3}. Paired native prefills instantiate the target-native statistics within closed-form mapper construction; stable linear solves then produce the affine parameters from unlabeled calibration statistics, keeping the mapper training-free.

Empty-history rows remain in Selection and Fit with their target supervision and processed weights. GQA/MQA sensitivity aggregates query heads through $h(g)$, and materialization writes the native target KV-head layout; the mapper preserves the native target KV-head count.

A.5 Complete Closed-Form Mapper Construction and Inference-Time Procedures

Algorithm A.1 gives the unified closed-form construction and inference-time procedure using the operators and metric rules above. Construction returns the frozen mapper $\theta_{S\to T} = \{\mathcal{I}_\ell, W^K_{\ell h}, c^K_{\ell h}, W^V_{\ell h}, c^V_{\ell h}\}_{\ell,h}$. Serialized layer selections `I0` use zero-based indices and preserve the returned top-k order.

Algorithm A.1 TACIT closed-form mapper construction and inference-time execution.

Closed-form mapper construction

`01` For each $x \in \mathcal{D}_{\text{cal}}$, tokenize both models with endpoints and native positions under explicit special-token control; run paired native prefills.

`02` Apply layer-specific DePos to both cache branches and construct $A_{\text{cal}} = \text{OneHotLastCausalGather}(e^S, e^T)$. Gather source features onto the target timeline.

`03` Extract strict-future ($i > j$) same-slot raw traces with `key_divide_by_dk=false`; append aligned source features, target content features, and traces.

`04` Stack calibration records into $S_{\text{all}}, T_{\text{all}}, m_{\text{raw}}$. Robustify and shrink weights independently for each branch, target layer, and KV head to form D.

`05` For each target layer, head-average the processed metrics into $\overline{D}_K, \overline{D}_V$. For each source candidate, project source/target prefix rows and slice matching selection weights.

`06` Compute weighted CKA scores γ_K, γ_V; set `I0` to the returned order of `torch.topk`$((\gamma_K + \gamma_V)/2, k)$.

`07` For each branch, assemble F in `I0` order and increasing source-head order. For each target KV head, take the matching target Y and processed metric D.

`08` Compute $\mu_f, \mu_y, F_c, Y_c, \lambda_{\text{eff}}$ as in section A.4.3; stably solve $(F_c^\top D F_c + \lambda_{\text{eff}} I)W = F_c^\top D Y_c$ and set $c = \mu_y - \mu_f W$.

`09` Freeze $\theta_{S\to T} = \{\texttt{I0}, W, c\}$ with its column order and store `fixed_cfg`.

Inference-time execution

`10` Tokenize x with endpoints and positions; set $n_c = n - R$ and split target tokens into $t_{1:n_c}$ and replay tail $r = t_{n_c+1:n}$. Run source-only prefill.

`11` Compact the union of source layers in frozen `I0`; batch layer-specific source DePos for each branch.

`12` Construct $A_x = \text{OneHotLastCausalGather}(e^S, e^T_{1:n_c})$ and batch-gather the compacted source features.

`13` Assemble target-layer features in frozen `I0` and source-head order. Stack independent affine maps into one batched GEMM per branch: $C_{\text{content}} = FW + c$.

`14` Vectorize target-layer RePos for K at $p^T_{1:n_c}$; leave V unchanged. Convert to target dtype and materialize contiguous per-layer caches in native GQA/MQA KV layout.

`15` Process replay tail r at $p^T_{n_c+1:n}$, then task suffix $q_{1:m}$ at subsequent positions, and generate $y_{1:o}$.

Construction uses the ordinary prefill consumers and raw-trace convention specified in section A.3; inference gathers exactly the n_c transported slots. Requests change endpoints and A_x, while $\mathcal{I}_\ell, W, c$ remain frozen. With contiguous zero-based positions, replay occupies $n_c, \ldots, n-1$ and task-suffix positions begin at n. Batched execution preserves target-layer, head, and branch identities and the frozen feature order.

B Experimental Protocols

B.1 Benchmarks and Scoring

CICERO-v1 follows the official `single_selection` protocol, selecting single-answer examples from 10,888 validation rows. ToMATO uses its complete released set, and Story Cloze uses `cloze_test_val__winter2018.csv`. Source-cluster counts are 1,097 for CICERO, 753 conversations for ToMATO, and 4,907 for SWAG.

Task instructions and answer cues join the target-native suffixes in table B.1; gold annotations are used for accuracy. After processing the context replay and task prompt, the target scores each complete candidate independently from the shared prefix state and selects the highest mean conditional log-likelihood over candidate tokens.

The cross-family comparisons in table 1 use the official HellaSwag validation split, pairing TACIT with Target Prefill in each direction (seed 117).

WikiText-103 uses the complete official `wikitext-103-raw-v1` test split, joining its 4,358 rows in order. Each window supplies 1,024 target context tokens followed by up to 256 non-

Benchmark	Evaluation set	Transported information	Target-native task suffix	Chance floor
HellaSwag	validation; 10,042 items	complete event context	four complete endings	25%
CICERO-v1	`val.json`; 9,470 items	dialogue context	question and five candidates	20%
ToMATO	all (`tomato.json`); 5,401 items	complete multi-turn dialogue	question and four mental-state candidates	25%
SWAG	validation (regular); 20,006 items	complete `startphrase`	four complete continuations	25%
PIQA	validation (plain_text); 1,838 items	complete physical `goal`	two solution candidates	50%
Story Cloze	Winter 2018 validation; 1,571 items	four-sentence story	two complete endings	50%

Table B.1: Complete labeled evaluation splits for table 3: per-direction sample counts, transported information, target-native suffixes, and chance floors.

overlapping continuation tokens, scored under teacher forcing. The source prefills the context; the target receives 1,023 mapped cache positions and replays the final boundary token.
SQuAD v1.1 uses all 10,570 development questions in each of four directions: Qwen3-8B→Llama-3.1-8B, Mistral-7B→Llama-3.2-3B, BLOOM-560M→Mistral-7B, and SmolLM3-3B→Mistral-7B. The target receives the mapped paragraph prefix, replays its final 32 tokens, and generates an answer greedily with a 32-new-token limit.

B.2 Calibration and Hyperparameters

All methods use 96 English documents per direction from the `codelion/finewiki-10M` snapshot of FineWiki (Penedo, 2025) for mapper construction or training, with source and target models frozen. For benchmark calibration, we use 256-target-token spans with stride-two sampling and eight additional documents for validation, yielding 12,288 fitting rows and 1,024 held-out rows per direction. TACIT uses a layer budget of $k = 2$, a default relative ridge ratio of $\rho = 10^{-4}$, and sensitivity shrinkage $\alpha = 0.5$, with normalized weights capped at 20. Sensitivity statistics use strictly subsequent queries $i > j$ within the same prefill. Weighted CKA uses 64-dimensional projections, with seeds 17 and 18 for the K and V branches, respectively. For TACIT's text continuation and answer generation, we select one shared $\rho = 10^{-3}$ by mean NLL across the shared validation directions on 128 FineWiki validation windows per direction, drawn from documents disjoint from mapper calibration and its held-out set.

Boundary replay. The shared replay length was selected on 24 independent PG-19 windows as the smallest length at most 32 within 0.01 nat/token of the best validation NLL. The complete-split C2C, MoT, LCF, LSC, and TACIT runs use the same context boundary per item and direction. For $N_T \geq 1$ shared-context tokens under the target tokenizer, the replay length is

$$R(N_T) = \min\{N_T - 1,\ 8,\ \max(1, \lfloor 0.05 N_T \rfloor)\}\,. \tag{B.1}$$

The mapper converts the first $N_T - R(N_T)$ positions; the target processes the final $R(N_T)$ context tokens natively before the task suffix. The length-proportional term limits replay on short contexts, while the eight-token cap bounds it on long contexts. The rule gives $R(1) = 0$.
The breadth study and ablation/mechanism studies use one-token replay for behavioral scoring. WikiText-103 and SQuAD follow their respective protocols in section B.1.

B.3 Mechanism and Mapper-Class Protocol

The primary mechanism case uses Qwen3-1.7B→Llama-3.2-3B-Instruct with the calibration settings above. Four held-out FineWiki documents provide 512 checkpoint-loss rows; the other four support attention diagnostics. For attention-guided GELU and Identity, held-out sensitivity weights are normalized and clipped across all eight documents before evaluating loss on the first four; training weights come from the calibration set. Behavioral evaluation uses the 512-item HellaSwag subset (seed 117), one-token replay, and candidate scoring defined above. Pointwise 95% intervals

use 10,000 paired item-bootstrap resamples (seed 118); p values use the exact two-sided McNemar test.

Native-trajectory attention measurements. Each passage supplies a 256-token native prefix and up to 64 following tokens (20–64 here, mean 46.5). At each layer, we replace prefix K, V, or both, retaining native queries, suffix K/V, and the unmodified prefix branch. Outputs are measured before W_O on queries strictly after the prefix. Relative L_2 is $\sqrt{\sum \|\widehat{O} - O\|_2^2 / \sum \|O\|_2^2}$, pooling the relevant output coordinates. Attention KL is $\mathrm{KL}(P_{\mathrm{native}} \| P_{\mathrm{mapped}})$ averaged over future query heads. These diagnostics fix each layer's native trajectory; HellaSwag interventions evaluate the complete target model.

The layerwise stage-change curves report $100(1 - E_{\mathrm{new}}/E_{\mathrm{preceding}})$, where E is that layer's native-to-mapped attention KL for K routing, or V-only attention-output relative L_2 for V read-out. Selection compares TACIT Selection with uniform Fit against Coordinate; Fit compares full TACIT against TACIT Selection with uniform Fit. Each percentage measures reduction from its preceding stage.

For fig. 7, relative target depth is $100(\ell - 1)/(L - 1)$ for one-based layer index ℓ, with $L = 28$ for Llama-3.2-3B and $L = 32$ for Llama-3.1-8B. Each point pools diagnostic queries and heads within that target layer using the metrics above. Plotting regions span 0–30%, 30–70%, and 70–100%; behavioral intervention blocks are specified in section C.

Scalar plots share native RMS normalization and asinh axes. Error ratios use full-tensor squared-error and native-energy sums.

C Supplementary Experiments

C.1 Transfer Performance and Coverage

C.1.1 Multitask Comparison

Table C.1: Fixed-method accuracies (%) by direction and benchmark.

Task	C2C	MoT	LCF	LSC	TACIT
	Qwen3-8B → Llama-3.1-8B				
HellaSwag	44.01	59.46	59.83	48.21	62.06
CICERO	20.36	31.12	31.73	27.77	35.02
ToMATO	30.75	50.08	52.92	42.71	53.84
SWAG	59.82	60.52	61.76	58.61	65.83
PIQA	64.25	62.08	60.12	55.88	64.36
Story Cloze	54.93	61.68	60.98	52.71	59.83
	Mistral-7B → Llama-3.2-3B				
HellaSwag	31.81	46.05	50.74	43.66	53.10
CICERO	15.66	25.46	24.89	25.61	32.20
ToMATO	28.79	33.77	38.72	30.81	49.06
SWAG	55.47	56.71	59.25	54.36	62.24
PIQA	64.80	54.35	59.63	58.54	64.85
Story Cloze	52.71	58.31	58.24	54.42	59.39
	BLOOM-560M → Mistral-7B				
HellaSwag	48.26	50.03	50.67	24.97	44.03
CICERO	26.07	30.82	34.36	12.46	36.43
ToMATO	30.57	39.71	42.40	24.90	38.42
SWAG	54.37	62.54	60.85	24.38	57.94
PIQA	71.38	71.44	73.01	53.37	72.85
Story Cloze	51.43	54.93	56.08	52.64	56.59
	Mistral-7B → Qwen3-8B				
HellaSwag	51.64	51.57	56.02	52.26	56.94
CICERO	28.34	32.29	33.99	31.36	33.37
ToMATO	49.38	40.59	44.42	40.46	45.86
SWAG	55.08	61.25	63.69	55.94	62.80
PIQA	71.44	66.10	68.66	70.73	72.47
Story Cloze	56.21	60.28	59.07	54.23	61.49
	SmolLM3-3B → Mistral-7B				
HellaSwag	50.26	63.53	60.86	25.16	63.56
CICERO	28.71	36.55	36.61	12.42	29.41
ToMATO	34.23	43.92	47.86	25.33	47.86
SWAG	55.28	67.59	65.74	24.85	66.94
PIQA	72.14	75.14	74.97	49.95	73.94

Table C.1 – continued from the previous page

Task	C2C	MoT	LCF	LSC	TACIT
Story Cloze	52.13	59.58	60.73	53.60	59.01
	Llama-3.2-3B → Qwen3-14B				
HellaSwag	58.03	61.98	63.06	56.41	62.46
CICERO	33.18	36.31	36.23	32.20	35.41
ToMATO	50.94	46.27	46.81	45.68	47.68
SWAG	59.11	67.57	68.03	58.08	66.33
PIQA	74.16	75.52	75.73	72.20	75.79
Story Cloze	56.59	64.16	63.14	55.82	63.21

Prefix	Native acc. / PPL	TACIT acc. / PPL	Coordinate acc. / PPL	No-history acc. / PPL
1K	61.5 / 6.22	54.7 / 8.82	50.0 / 13.49	40.6 / 44.57
4K	62.5 / 4.97	51.6 / 9.98	44.3 / 20.71	42.2 / 38.84
16K	59.9 / 4.82	56.2 / 6.94	45.3 / 14.35	46.9 / 29.92
32K	69.3 / 3.26	62.0 / 5.48	52.1 / 9.55	44.8 / 24.50

Table C.2: First 32 continuation tokens after handoff. Each cell reports next-token accuracy (%) / PPL.

C.2 Construction and Deployment Efficiency

C.2.1 Matched Single-GPU Construction

We evaluate mapper construction on one 80-GiB H800 with three fresh-process trials per direction. Documents contain 256–2048 tokens under every tokenizer and are processed in full. Closed-form construction samples KV rows at stride two; learned methods score the final 64 target tokens using the preceding prefix.

Timing boundary. We measure mapper construction from paired-KV artifact loading through mapper saving, including initialization, optimization, and holdout stopping for learned methods. TACIT's path includes attention-statistic forward passes, Selection, closed-form Fit from stored KV and weights, and serialization. We also report the complete construction path with one shared KV extraction.

Aggregation and memory. Each direction contributes its median; tables average the direction-level medians. One-GPU wall minutes equal reserved GPU-minutes. Memory is sampled every 0.5 seconds above idle usage. Full peak memory is the maximum of extraction and method-stage peaks; Worst is the largest trial peak.

	Time (min/mapper)		Peak memory (GiB)		
Method	Method	Full	Method mean	Full mean	Worst
C2C	5.65	8.60	45.32	45.32	58.42
MoT	4.38	7.33	39.04	39.04	51.94
LCF	3.12	6.07	33.91	33.91	44.45
LSC	7.54	10.49	33.97	33.97	43.75
TACIT	2.21	5.16	29.47	29.97	44.78

Table C.3: Single-GPU construction costs, averaged over direction-level medians. Method reports mapper construction, while Full reports the complete construction path with shared extraction. Worst reports the maximum full-construction peak.

C.2.2 Mapper Parameters and Storage

The frozen mapper contains a separate affine map for each target layer, KV head, and K/V path. For the evaluated models, the per-head K and V widths coincide, so its weight-and-bias parameter count is

$$P = 2L_T H_{\mathrm{KV},T} d_T \big(k H_{\mathrm{KV},S} d_S + 1\big), \qquad k = 2. \tag{C.1}$$

The evaluated directed mappers contain 67.17–268.57 million affine parameters. Table C.5 reports their parameter counts and single-copy affine tensor payloads, which equal $4P$ bytes in FP32. The

Direction	C2C	MoT	LCF	TACIT
Qwen3-8B → Llama-3.1-8B	6.33 / 50.22	4.25 / 44.83	3.42 / 39.74	2.01 / 26.79
Mistral-7B → Llama-3.2-3B	4.79 / 37.75	4.34 / 31.91	3.12 / 28.25	1.54 / 14.77
BLOOM-560M → Mistral-7B	4.81 / 35.00	3.43 / 30.10	2.61 / 23.69	2.58 / 30.73
Mistral-7B → Qwen3-8B	7.72 / 52.44	4.54 / 46.18	3.16 / 40.26	2.71 / 29.00
SmolLM3-3B → Mistral-7B	4.71 / 38.10	3.91 / 29.30	2.80 / 27.05	1.98 / 30.73
Llama-3.2-3B → Qwen3-14B	5.53 / 58.42	5.81 / 51.94	3.62 / 44.45	2.44 / 44.78

Table C.4: Matched single-GPU method-stage construction by direction. Cells give minutes / peak GiB, each the median over trials.

mean payload is 490.93 MiB. For Qwen3-8B→Llama-3.1-8B, $d_F = 2048$, $N/d_F = 6$, and the FP32 affine payload is 512.25 MiB.

Table C.5: Mapper parameters and storage for the evaluated cross-family directions with $k = 2$. Parameters include K/V weights and biases; payload counts one FP32 tensor copy at four bytes per parameter.

Direction	Params (M)	Tensor payload (MiB)
Qwen3-8B→Llama-3.1-8B	134.28	512.25
Mistral-7B→Llama-3.2-3B	117.50	448.22
BLOOM-560M→Mistral-7B	134.28	512.25
Mistral-7B→Qwen3-8B	151.07	576.28
SmolLM3-3B→Mistral-7B	67.17	256.25
Llama-3.2-3B→Qwen3-14B	167.85	640.31
Qwen3-4B→Llama-3.2-3B	117.50	448.22
Mistral-7B→Qwen3-14B	167.85	640.31
Llama-3.1-8B→Qwen3-32B	268.57	1024.50
Qwen3-1.7B→Llama-3.2-3B	117.50	448.22
Qwen3-1.7B→Mistral-7B	134.28	512.25
BLOOM-560M→Llama-3.2-3B	117.50	448.22
Llama-3.2-3B→BLOOM-560M	100.71	384.19

C.2.3 Mapper Capacity and Cost

Table 5 varies $k \in \{1, 2, 4\}$ on Qwen3-8B→Llama-3.1-8B and Mistral-7B→Qwen3-8B. Accuracy averages the two directions; 95% intervals use 10,000 percentile-bootstrap resamples paired across directions and k settings (seed 94217), holding mappers fixed. Latency pools three fresh processes per direction, each using 16 deterministic prefix-length quantiles, two untimed initialization passes, and five timed repetitions. The table reports resident-cache handoff p50 and p90, averaged over direction-level percentiles. Doubling k from 2 to 4 halves the calibration-row-to-feature ratio N/d_F from 6 to 3 at the fixed data budget.

C.2.4 Benchmark-Conditioned Cache Handoff

The benchmark-conditioned panel in table 3 reports cache conversion and delivery separately from mapper construction. It includes alignment, mapping, target-KV materialization, and cross-GPU delivery; source prefill, model loading, candidate scoring, and decoding are excluded.

Batched cache delivery. TACIT packs K and V into contiguous slabs for two total NVLink copies, giving a mean handoff of 6.13 ms. Replay follows eq. (B.1).

C.2.5 Prefix-Length Handoff Scaling

For Qwen3-8B→Llama-3.1-8B, table 2 uses six FineWiki long articles per length and 256 teacher-forced continuation tokens per document. Quality is next-token accuracy on the natural continuation; its intervals use document-level bootstrap resampling. Native TTFT is target prefill, while TACIT TTFT includes resident-cache conversion, the required delivery, and one native boundary-token forward pass. Timing uses BF16 SDPA on H800 GPUs, with source and mapper on GPU 0 and target on GPU 1; source prefill is resident and excluded.

Prefix	Native prefill (ms)	TACIT handoff (ms)	Replay (ms)	TACIT TTFT (ms)	TTFT speedup	Handoff-only speedup	Coordinate handoff (ms)	Source prefill (ms)
1K	32.8	5.7	15.9	21.6	1.52×	5.74×	5.8	40.0
4K	131.9	19.6	16.0	35.7	3.70×	6.73×	19.8	156.5
16K	672.8	75.4	15.9	91.3	7.37×	8.93×	76.1	789.2
32K	1762.3	152.1	16.0	168.2	10.48×	11.59×	153.7	2160.1

Table C.6: Prefix-length latency decomposition for the quality/TTFT scaling study. TTFT equals handoff plus boundary replay; source prefill is excluded.

C.3 Functional Geometry and Mapper Design

C.3.1 Target-Local Attention Metric

The local experiment covers Qwen3-8B→Llama-3.1-8B, Mistral-7B→Llama-3.2-3B, and the Llama-3.2-3B→Llama-3.1-8B control. Operator validation uses a separate FineWiki diagnostic set, eight uniformly selected target layers, and all eight target KV heads. The target-native query is fixed at position 255. Measurements act on the target-attention operator before the output projection at a fixed query, layer, and head.

For the perturbation sweep $\epsilon \in \{0.25, 0.5, 1, 2\}$, support is fixed to slots $\{0, 2, \ldots, 254\}$ strictly before the current query position; the current query position is not perturbed. The projection bank crosses the four Selection×Fit cells with identity, slot shuffle, signed channel permutation, and joint slot-and-channel transformations. Each method residual is first normalized to the within-group median coordinate norm and then scaled by the shared $\{0.75, 1, 1.25\}$ norm ladder. Each fixed direction×document×layer×head×branch×ϵ group therefore contains 48 perturbations. Correlations are computed within these fixed groups. Every direction–branch–ϵ entry in table C.7 is then the median over 512 document–layer–head groups. K and joint fidelity decrease as perturbation scale grows, whereas V remains nearly exactly linear at every scale, with $R^2 = 1.0000$. Figure 4 pools the group-level correlations and reports their median, with 95% percentile intervals from document-cluster bootstrap resampling within each direction.

Direction	Branch	$\epsilon = 0.25$	0.5	1	2
Qwen→Llama	K	0.9900	0.9700	0.8932	0.6859
	V	0.9994	0.9995	0.9996	0.9997
	joint	0.9926	0.9798	0.9452	0.8831
Mistral→Llama	K	0.9917	0.9726	0.8768	0.6111
	V	0.9994	0.9995	0.9996	0.9997
	joint	0.9940	0.9844	0.9471	0.7640
Llama→Llama control	K	0.9928	0.9771	0.9223	0.7642
	V	0.9993	0.9994	0.9995	0.9996
	joint	0.9941	0.9853	0.9602	0.9149

Table C.7: Median Spearman correlation between the full pullback geometry $J_T^\top J_T$ and nonlinear attention-output distortion.

We use the prespecified $\epsilon = 0.25$ and form joint energy as $J_K e_K + J_V e_V$ before squaring to retain the K/V cross-term.

Mean-trace exceeds coordinate distance in both Spearman correlation and R^2 for every direction×branch panel. It projects each retained same-slot block to a scalar weight, while the same-slot anisotropic comparator preserves channel structure and serves as the richer reference.

For the fitted-mapper experiment, the distortion ratio is evaluated at the canonical $\epsilon = 1$. Under the same fixed Selection condition, it divides the nonlinear target-attention distortion of mean-trace-conditioned Fit by that of coordinate-conditioned Fit; a value below 1 favors mean-trace Fit.

C.3.2 Selection and Fit

Coordinate Selection uses unit weights for layer ranking; Coordinate Fit sets $D^b_{\ell h} = I$. Crossing each with its attention-derived counterpart isolates the geometry used for Selection and Fit.

All four cells share the affine function class, calibration data, source-layer budget, alignment, position handling, and runtime interface. In table C.10, the Selection effect is mean-trace Selection minus coordinate Selection at fixed Fit; the Fit effect is mean-trace Fit minus coordinate Fit at fixed Selection. The interaction is the Selection effect under mean-trace Fit minus that under coordinate Fit.

Branch	Coord.		Mean-trace		Same-slot aniso.	
	ρ	R^2	ρ	R^2	ρ	R^2
Qwen3-8B→Llama-3.1-8B						
K	0.3682	0.1202	0.7544	0.4267	0.9727	0.9743
V	0.4373	0.1320	0.9971	0.9999	0.9971	0.9999
joint	0.4059	0.1344	0.9795	0.9974	0.9957	0.9996
Mistral-7B→Llama-3.2-3B						
K	0.3438	0.0967	0.5517	0.2242	0.9656	0.9774
V	0.4233	0.1803	0.9979	0.9998	0.9979	0.9998
joint	0.4253	0.2022	0.9626	0.9501	0.9947	0.9963
Llama-3.2-3B→Llama-3.1-8B control						
K	0.3922	0.1226	0.8081	0.4677	0.9745	0.9672
V	0.4441	0.1236	0.9969	0.9999	0.9969	0.9999
joint	0.4173	0.1260	0.9803	0.9982	0.9950	0.9997

Table C.8: Metric projection at $\epsilon = 0.25$, reported as medians over the fixed document–layer–head groups.

Table C.9: Nonlinear attention-output distortion ratios for mean-trace Fit relative to coordinate Fit under two fixed shared-Selection conditions.

Direction	Branch	Geometric mean ratio	Document-bootstrap 95% log-ratio CI
Fixed Selection: coordinate			
Qwen3-8B→Llama-3.1-8B	joint	0.5254	$[-0.8486, -0.4226]$
	K	0.7013	$[-0.4790, -0.2257]$
	V	0.5101	$[-0.8711, -0.4758]$
Mistral-7B→Llama-3.2-3B	joint	0.4951	$[-0.9824, -0.4465]$
	K	0.5498	$[-1.4000, 0.1930]$
	V	0.6077	$[-0.7621, -0.2683]$
Llama-3.2-3B→Llama-3.1-8B control	joint	0.7032	$[-0.4531, -0.2655]$
	K	0.8024	$[-0.3179, -0.1274]$
	V	0.6999	$[-0.4590, -0.2735]$
Fixed Selection: attention			
Qwen3-8B→Llama-3.1-8B	joint	0.6268	$[-0.6370, -0.3103]$
	K	0.9465	$[-0.1808, 0.0382]$
	V	0.5836	$[-0.7088, -0.3826]$
Mistral-7B→Llama-3.2-3B	joint	0.2944	$[-1.5311, -0.8491]$
	K	0.2614	$[-2.1872, -0.5166]$
	V	0.4658	$[-0.9658, -0.5590]$
Llama-3.2-3B→Llama-3.1-8B control	joint	0.7155	$[-0.4365, -0.2517]$
	K	0.8252	$[-0.2918, -0.0997]$
	V	0.7088	$[-0.4457, -0.2628]$

Fit raises accuracy across fixed-Selection contrasts, while Selection supplies its largest gains on the Qwen-source directions. The interaction terms show that the two components act as coupled functional choices.

C.3.3 Mapper Function Class

The function-class comparison uses Qwen3-1.7B→Llama-3.2-3B-Instruct under the protocol in section B.3. Within each geometry, the affine and learned controls share Selection, calibration rows, dimensions, and reconstruction weights. Coordinate geometry uses uniform weights, whereas attention-guided geometry uses target-attention weights. The affine map uses closed-form ridge; learned maps use AdamW with held-out checkpoint selection.

(a) Complete Selection×Fit accuracies (%)

Direction	Coord. Sel. + Coord. Fit	Mean-trace Sel. + Coord. Fit	Coord. Sel. + Mean-trace Fit	Full TACIT
Qwen3-8B→Llama-3.1-8B	45.90	62.11	52.73	62.89
Mistral-7B→Llama-3.2-3B	33.40	31.45	54.30	55.27
Qwen3-4B→Llama-3.2-3B	38.28	53.32	56.25	56.64
Mistral-7B→Qwen3-8B	52.73	50.98	58.59	58.20

(b) Paired gains and Selection effects (pp)

Direction	Full−Coord.	Sel. @ coord. Fit	Sel. @ mean-trace Fit
Qwen3-8B→Llama-3.1-8B	$+16.99\,[+13.28, +20.70]$	$+16.21\,[+12.50, +19.92]$	$+10.16\,[+7.03, +13.48]$
Mistral-7B→Llama-3.2-3B	$+21.88\,[+17.38, +26.17]$	$-1.95\,[-5.47, +1.56]$	$+0.98\,[-1.37, +3.32]$
Qwen3-4B→Llama-3.2-3B	$+18.36\,[+14.06, +22.66]$	$+15.04\,[+10.94, +19.14]$	$+0.39\,[-1.95, +2.73]$
Mistral-7B→Qwen3-8B	$+5.47\,[+2.93, +8.20]$	$-1.76\,[-4.11, +0.59]$	$-0.39\,[-2.34, +1.56]$

(c) Fit effects and interactions (pp)

Direction	Fit @ coord. Sel.	Fit @ mean-trace Sel.	Interaction
Qwen3-8B→Llama-3.1-8B	$+6.84\,[+3.91, +9.96]$	$+0.78\,[-0.78, +2.34]$	$-6.05\,[-9.57, -2.54]$
Mistral-7B→Llama-3.2-3B	$+20.90\,[+16.60, +25.20]$	$+23.83\,[+19.14, +28.52]$	$+2.93\,[-1.17, +7.03]$
Qwen3-4B→Llama-3.2-3B	$+17.97\,[+13.87, +22.27]$	$+3.32\,[+0.98, +5.86]$	$-14.65\,[-19.14, -10.35]$
Mistral-7B→Qwen3-8B	$+5.86\,[+3.32, +8.59]$	$+7.23\,[+4.30, +10.16]$	$+1.37\,[-1.56, +4.30]$

Table C.10: Selection/Fit results on the paired HellaSwag evaluation. Brackets give paired-bootstrap 95% intervals. Each effect holds the other component fixed; interaction is the difference between the two Selection effects.

The factorized maps use $2048 \to 682 \to 1024$ bottlenecks across 28 layers; each branch has 58,710,680 learned parameters versus 58,748,928 for the affine branch (99.935%). Identity replaces GELU at fixed parameter count, and both bottlenecks have output dimension at most 682.

Geometry / mapper	Full-split Acc. (%)
Coordinate / Full affine	25.83
Coordinate / Factorized linear	27.79
Coordinate / GELU MLP	36.51
Attention-guided / Full affine	53.74
Attention-guided / Factorized linear	51.38
Attention-guided / GELU MLP	50.35

Table C.11: Parameter-matched mapper controls on the full HellaSwag validation split.

C.4 Mechanistic Analysis

Representative depth profiles. Figure C.1 complements the two main-text Qwen-to-Llama cases with two different branch and depth patterns. For BLOOM-560M→Mistral-7B, Fit reduces aggregate attention KL by 17.30%, while V-only output error under native attention falls by 1.09%. For Llama-3.2-3B→Qwen3-14B, the corresponding reductions are 22.18% and 20.46%, with V-only error decreasing across all 40 target layers. Joint K+V output error falls by 35.90% and 31.13%, respectively (table C.14). Relative depth is $100(\ell-1)/(L-1)$, with each target's layer count shown in the figure.

Branch and layer interventions. Selection controls replace coordinate-fitted maps in the indicated branch or target block; Fit controls replace uniform weights with attention-weighted fits. Early, middle, and late target blocks are layers 1–9, 10–18, and 19–28, and block interventions replace both K and V. V-only Selection spans all layers, whereas early-only Selection changes both branches in its block.

Functional effects and sensitivity-position controls. Table C.13 pairs operator measurements with behavioral contrasts under fixed TACIT Selection.

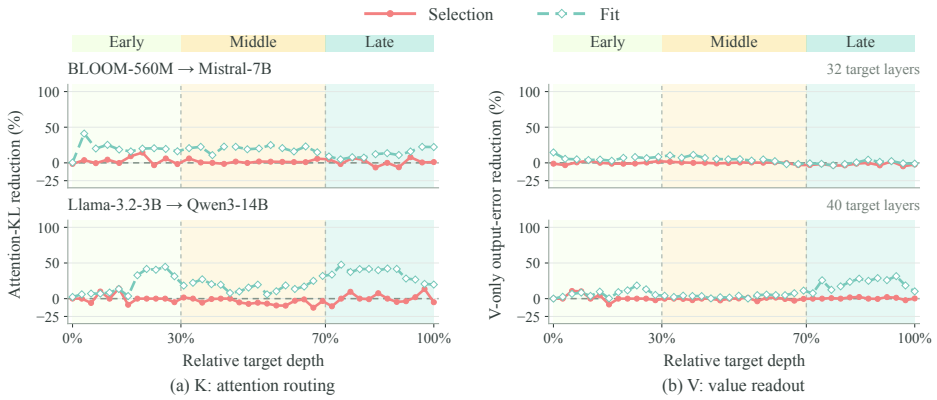

Figure C.1: Forward profiles: BLOOM-560M→Mistral-7B and Llama-3.2-3B→Qwen3-14B. (a) Attention-KL and (b) V-only output-error reduction. Selection uses Coordinate as its baseline; Fit uses uniform fitting with TACIT Selection fixed. Positive is better. The 30%/70% plotting boundaries differ from layer-block interventions.

Intervention	Acc. (%)	Δ (pp)	Paired 95% CI	p
TACIT Selection; coordinate Fit throughout; vs. all-coordinate				
K branch, all layers	27.93	$+1.17$	$[-0.98, +3.52]$	0.392
V branch, all layers	53.91	$+27.15$	$[+22.46, +31.84]$	$< 10^{-20}$
Early layers, K+V	54.10	$+27.34$	$[+22.66, +32.23]$	$< 10^{-20}$
Middle layers, K+V	25.59	-1.17	$[-2.93, +0.59]$	0.286
Late layers, K+V	27.34	$+0.59$	$[-0.20, +1.56]$	0.375
Attention Fit; TACIT Selection fixed; vs. uniform Fit				
Early layers, K+V	55.08	$+1.56$	$[+0.20, +3.13]$	0.0574
Middle layers, K+V	53.32	-0.20	$[-1.37, +0.98]$	1.000
Late layers, K+V	54.10	$+0.59$	$[-0.39, +1.56]$	0.453

Table C.12: Paired branch and layer interventions on the HellaSwag evaluation. Each block states its fixed component and comparison baseline.

All controls hold TACIT Selection and affine capacity fixed. True sensitivities are normalized per target layer and KV head, clipped at 20 times their mean, and mixed equally with uniform weights. Shuffling preserves per-document weight distributions; anchor variants alter only the first regression row. Every condition is refitted in closed form.

Fitting weights	Acc. (%)	Output rel. L_2	Attn. KL	Ref.	Δ (pp) [paired 95% CI]	p
Uniform (U)	53.52	0.385	0.159	—	—	—
True (T)	55.86	0.376	0.112	U	$+2.34$ $[+0.59, +4.10]$	0.0169
K only	53.91	0.381	0.112	U	$+0.39$ $[-0.78, +1.56]$	0.754
V only	54.49	0.381	0.159	U	$+0.98$ $[-0.39, +2.34]$	0.267
All-position shuffled	53.91	0.391	0.162	T	-1.95 $[-3.91, -0.20]$	0.0639
Anchor only (A)	54.30	0.381	0.160	U	$+0.78$ $[-0.20, +1.95]$	0.289
Anchor emphasis removed	55.47	0.378	0.110	U	$+1.95$ $[+0.59, +3.52]$	0.0213
Non-anchor shuffled	54.10	0.386	0.165	T	-1.76 $[-3.52, -0.20]$	0.0636
Additional contrasts for the same conditions						
Anchor emphasis removed	—	—	—	T	-0.39 $[-1.56, +0.59]$	0.727
True	—	—	—	A	$+1.56$ $[0.00, +3.13]$	0.0963

Table C.13: Functional and behavioral effects of fitting under fixed TACIT Selection. Output error replaces prefix K/V; KL compares native and mapped attention. Contrasts use uniform (U), true sensitivity (T), or anchor-only (A) references; p is the exact two-sided McNemar value.

Transfer direction	Coord. Acc.	Selection Acc.	+ Fit Acc.	Fit gain (pp) [95% CI]	Joint output rel. L_2 Uniform → Fit
Qwen3-8B → Llama-3.1-8B	45.90	62.11	63.48	$+1.37$ $[0.00, 2.93]$	$0.380 \rightarrow 0.361$
Mistral-7B → Llama-3.2-3B	33.40	31.25	55.47	$+24.22$ $[19.73, 28.71]$	$0.941 \rightarrow 0.544$
BLOOM-560M → Mistral-7B	37.70	35.16	46.09	$+10.94$ $[7.23, 14.84]$	$0.961 \rightarrow 0.616$
Mistral-7B → Qwen3-8B	52.73	50.98	58.01	$+7.03$ $[4.10, 9.96]$	$1.579 \rightarrow 1.129$
SmolLM3-3B → Mistral-7B	58.59	58.40	63.67	$+5.27$ $[2.54, 8.01]$	$0.941 \rightarrow 0.527$
Llama-3.2-3B → Qwen3-14B	62.89	63.48	65.43	$+1.95$ $[0.20, 3.71]$	$0.439 \rightarrow 0.302$

Table C.14: Mechanism-stage results across the evaluated forward directions. Accuracies are percentages. Selection changes source layers with coordinate Fit retained; + Fit adds attention-aware fitting. Fit gains report paired 95% CIs. Joint error measures K+V prefix replacement before the output projection, using native queries on four held-out passages; both gains and errors compare uniform Fit with attention-aware Fit.

Early-layer and value-branch Selection. On Qwen3-8B→Llama-3.1-8B, applying TACIT Selection to V alone raises accuracy from 45.90% to 62.50% ($+16.60$ pp, paired 95% CI $[12.89, 20.31]$). Applying Selection to both branches in the early target-layer third (layers 1–10 of 32) gives 62.70% ($+16.80$ pp, $[13.28, 20.31]$), close to the 62.11% obtained with Selection throughout.

Attention-aware Fit and sensitivity placement. On Mistral-7B→Llama-3.2-3B, with TACIT Selection fixed, attention-aware Fit raises accuracy from 31.25% to 55.47% and reduces joint attention-output relative L_2 by 42.24%. The distribution-matched weight shuffle provides a 53.32% control, confirming that the useful signal is the association between sensitivity weights and cache positions.

C.5 Robustness and Sensitivity

Table C.15 summarizes shrinkage, calibration-seed, and projection-seed controls. All controls follow section B; ensemble CKA is averaged before top-k selection, with seed 17 as the reference.

(a) Sensitivity shrinkage; reference $\alpha = 0.5$

α	Accuracy (%)	Δ vs. 0.5 (pp)	Paired 95% CI
0.25	62.70	-0.20	$[-1.95, +1.56]$
0.50	62.89	0.00	$[0.00, 0.00]$
0.75	62.11	-0.78	$[-2.54, +0.98]$
1.00	61.33	-1.56	$[-3.91, +0.78]$

(b) Calibration seeds

Calibration seed	17	29	43
Qwen3-8B→ Llama-3.1-8B	45.90 → 62.89 $+16.99$ [13.28, 20.70]	37.70 → 64.06 $+26.37$ [22.07, 30.86]	51.76 → 63.09 $+11.33$ [8.20, 14.45]
Mistral-7B→ Llama-3.2-3B	33.40 → 55.27 $+21.88$ [17.38, 26.17]	35.55 → 53.91 $+18.36$ [14.06, 22.66]	33.20 → 52.15 $+18.95$ [14.26, 23.63]
Mistral-7B→ Qwen3-8B	52.73 → 58.20 $+5.47$ [2.93, 8.20]	55.27 → 56.45 $+1.17$ $[-1.56, 3.91]$	50.59 → 57.42 $+6.84$ [3.71, 9.96]

(c) Projection seeds and selected-layer stability

Projection seed	Accuracy (%)	TACIT−Coordinate (pp) [95% CI]	Mean layer overlap (%)	Mean Jaccard (%)
17 (default)	62.89	$+16.99$ $[+13.28, +20.70]$	100.0	100.0
29	60.94	$+15.04$ $[+11.33, +18.75]$	43.8	33.3
43	62.89	$+16.99$ $[+13.28, +20.70]$	43.8	32.3
Ensemble 17/29/43	62.89	$+16.99$ $[+13.28, +20.70]$	65.6	55.2

Table C.15: Sensitivity controls. Accuracy is in percent; differences are in percentage points with paired 95% intervals. Panel (a) compares shrinkage values with the default. Panel (b) gives Coordinate → TACIT accuracy above its paired gain; panel (c) also compares TACIT with Coordinate.