DOPAMINE: BRAIN MODES, NOT BRAINS*

Anonymous authors
Paper under double-blind review

ABSTRACT

Personalization may be represented at the wrong level. A common design stores a user as a change to model weights; we ask whether one frozen language model can instead express many user-conditioned modes, selected by small activation states. We test this hypothesis with DOPAMINE, one learned additive residual vector per transformer layer, using an operator with precedents in steering and bias-only adaptation. Across three model families from 0.36B to 14B parameters, controlled preferences, and Amazon-review histories, these modes achieve personalization quality comparable to LoRA-$r8$ with 140–215$\times$ fewer per-user parameters. On synthetic preferences, both methods average 0.876 accuracy over three seeds; across seven model/scale cells, their largest absolute gap is 0.012. On rating prediction, DOPAMINE obtains 0.732 versus the best tested LoRA's 0.739 accuracy at 0.36B, and 0.728 versus LoRA-$r8$'s 0.713 at 1.5B, with lower held-out cross-entropy in both cases. The representation permits row-wise user conditioning in a shared dense forward. Two negative results bound the claim: tested multiplicative states fail to improve generation accuracy, and a 64-dimensional PCA code fails despite retaining 96% of mode-vector variance. The evidence supports reconsidering the personalization interface: user-specific behavior can often be selected through activation state without user-specific weight transformations.

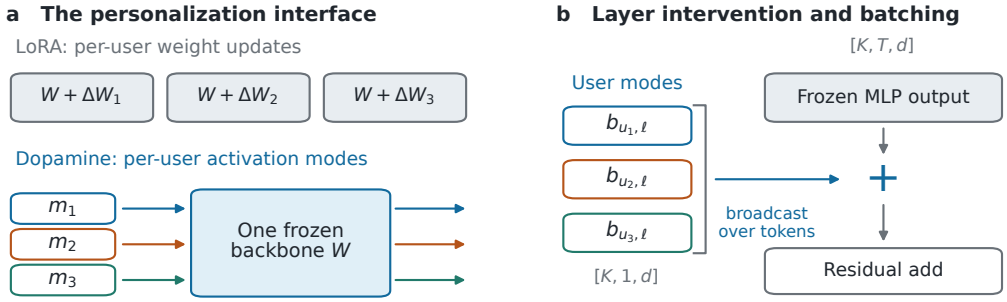


Figure 1: **Are we representing personalization at the wrong level?** (a) LoRA stores user-specific weight updates; DOPAMINE supplies activation modes to one frozen backbone. Both store learned parameters; the distinction is *where the user-specific information acts*. (b) At each layer, each batch row selects its user vector, which broadcasts across token positions and is added to the MLP output before residual addition. Shapes use batch size K, sequence length T, and width d. This illustrates the operator and row-wise composition, not a measured throughput result.

1 INTRODUCTION

Personalization may be represented at the wrong level. When a language model must serve many users, a common adaptation strategy assigns each user a parameter update: a different effective

*The words *brain* and *mode* are functional metaphors for a shared parameterized system and a user-conditioned activation intervention. DOPAMINE is a method name, not a model of dopamine or biological neuromodulation. We make no anthropomorphic claim about cognition, consciousness, agency, or biological equivalence.

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.

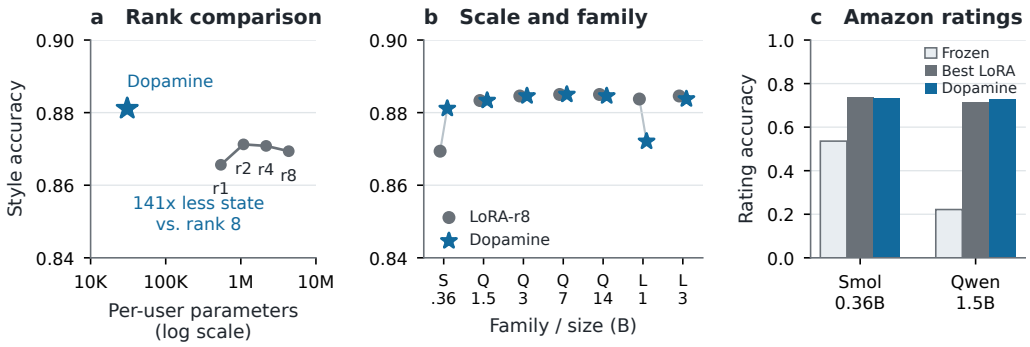


Figure 2: **The central evidence, before the implementation details.** (a) On the archived SmolLM2-360M seed-0 cells, a 30.7K-parameter DOPAMINE mode attains comparable or higher style accuracy than LoRA ranks 1–8 (0.54–4.34M parameters). (b) Across seven model/scale cells, its absolute gap from LoRA-$r8$ is at most 0.012. S, Q, and L denote SmolLM2, Qwen2.5, and Llama-3.2; sizes are in billions. (c) On Amazon rating prediction, accuracy is close to the best tested LoRA rank at 0.36B and higher at 1.5B. These are observed cell means, not statistical-equivalence claims; Tables 1–4 give the numbers and comparison scope. Panel (a) uses a logarithmic parameter axis and a restricted vertical range to expose the rank comparison.

model for each person. Parameter-efficient fine-tuning makes these updates smaller, but leaves the representational choice intact. We ask a prior question: *does personalized behavior require user-specific weight transformations, or can it be selected by a small mode of one shared, frozen model?*

The distinction is operational. A weight update changes the user-specific linear transformations applied throughout inference. An activation mode instead provides a small conditioning state to a fixed set of transformations. This state is learned, persistent, and user-specific; it is neither a transient key–value cache nor a claim that personalization requires no trainable parameters. Figure 1 makes the abstraction explicit.

We instantiate the hypothesis with DOPAMINE. For each user, it learns one constant residual vector per transformer layer and adds that vector at every token position. All backbone weights remain frozen, and zero initialization recovers the original model exactly. The state contains only Ld_{model} scalars: 30,720 for SmolLM2-360M and 245,760 for Qwen2.5-14B. There is no user-specific attention matrix, MLP matrix, or low-rank matrix product in the forward operator.

The primitive is intentionally simple and has close precedents. Activation steering changes model behavior through hidden representations (Subramani et al., 2022; Turner et al., 2023); representation fine-tuning learns such interventions (Wu et al., 2024); personalized steering and user embeddings already use compact signals for behavioral control (Cao et al., 2024; Doddapaneni et al., 2024). Most directly, Sinii et al. (2025) learn a single additive vector per layer for reasoning. We do not claim the additive operator, or the general idea of state-conditioned personalization, as a new invention. Our contribution is an empirical test of its sufficiency as the primary per-user adaptation interface, against a widely used weight-adaptation baseline.

The results make that test consequential (Figure 2). On controlled synthetic preferences, DOPAMINE and LoRA-$r8$ both average 0.876 accuracy over three seeds. Across three model families and seven model/scale cells spanning 0.36B–14B, their largest absolute difference is 0.012. DOPAMINE uses 140–215$\times$ fewer per-user parameters than LoRA-$r8$. On Amazon-review histories, it achieves 0.732 versus the best tested LoRA's 0.739 rating accuracy at 0.36B, and 0.728 versus 0.713 at 1.5B. Its held-out cross-entropy is lower in both comparisons.

Two negative results limit the interpretation. Multiplicative activation gates can reduce teacher-forced loss without materially improving the generation metric. Conversely, compressing additive modes into a shared 64-dimensional PCA code destroys their personalization performance despite retaining

96% of vector variance. Compact state is therefore not automatically sufficient: the intervention family and the behavioral information preserved both matter.

Our contributions are: (i) a precise *representational-level hypothesis* for personalization, separating per-user state from shared computation; (ii) an evaluation of a minimal additive mode against LoRA across model scale, family, and controlled-preference and Amazon-review tasks; and (iii) an account of its boundaries, including unsuccessful gating and compression, and a row-wise composition property with a clear distinction between algebra and measured serving performance. The broader thesis is deliberately conditional: *before optimizing the cost of separate effective models, we should test whether users need them.*

2 Related work

Weight-space and prompt-space adaptation. Adapters learn small modules inserted into a frozen network (Houlsby et al., 2019). LoRA parameterizes updates to weight matrices as low-rank products (Hu et al., 2022); BitFit restricts training to biases (Ben Zaken et al., 2022); and IA^3 rescales intermediate activations (Liu et al., 2022). Prompt tuning and prefix tuning place learned vectors in the input or attention context (Lester et al., 2021; Li & Liang, 2021). These methods already demonstrate that downstream behavior need not require full-model training. Our comparison asks a more specific question about the interface used to store many individual users. LoRA is the main empirical baseline here, not an exhaustive proxy for all PEFT methods.

Steering and representation fine-tuning. Latent steering vectors (Subramani et al., 2022), activation addition (Turner et al., 2023), and function vectors (Todd et al., 2024) show that hidden-state interventions can select behavior. ReFT formalizes learned representation interventions and evaluates them across downstream tasks (Wu et al., 2024). BiPO learns steering vectors through preference optimization for personalized behavioral control (Cao et al., 2024). These are direct conceptual precedents, not unrelated efficiency baselines. Bias-only reasoning adaptation (Sinii et al., 2025) is particularly close at the operator level: one learned d-dimensional vector per layer with fixed backbone weights. We use supervised per-user histories rather than its reasoning-training setting, and evaluate the resulting state as an alternative to per-user LoRA. The evidence concerns this application and comparison; it does not establish a new general class of intervention.

Personalization and user representations. LaMP evaluates language models conditioned on user profiles (Salemi et al., 2024). User embedding models compress histories into soft prompts (Doddapaneni et al., 2024), illustrating that users can already be represented as compact conditioning signals. Our Amazon experiment is a *LaMP-3-style rating task* constructed from Amazon Reviews 2023 (Hou et al., 2024), not an evaluation on the official LaMP split. It measures whether independently learned activation modes retain useful user information under a fixed adaptation protocol. We do not compare against all history encoders, retrieval-based personalization, or prompt-based systems.

Many-user serving. S-LoRA and Punica batch and serve distinct LoRA adapters using specialized systems (Sheng et al., 2024; Chen et al., 2023). Thus, shared-backbone serving is not unique to activation modes, and LoRA does not require keeping one full backbone copy per user. The narrower distinction is that an additive mode enters as a broadcast addition, without a user-conditioned matrix multiplication. This makes mixed-user composition simple to express. It does not by itself prove higher throughput than an optimized multi-adapter server.

3 Personalization as a mode of shared computation

3.1 A representational-level hypothesis

Let W be the shared backbone weights, $\mathcal{D}_u$ the adaptation history for user u, and x a new input. A weight-space method learns $\Delta W_u = \mathcal{A}_W(\mathcal{D}_u)$ and evaluates $p(y \mid x, u) = f(x; W + \Delta W_u)$. For LoRA, each adapted matrix has $\Delta W_u = (\alpha/r) B_u A_u$. The mode formulation instead learns $m_u = \mathcal{A}_m(\mathcal{D}_u)$ and uses

$$p(y \mid x, u) = f_W(x; m_u), \qquad W \text{ fixed across users.} \tag{1}$$

The hypothesis is that a suitably chosen, small m_u can retain much of the personalization obtainable through ΔW_u on the tasks considered. It is not a statement that a user has a unique, universal latent identity.

The two formulations need not define disjoint function classes. In particular, a constant activation addition can be absorbed into a corresponding bias when such a bias is present. DOPAMINE can therefore also be described as per-user bias adaptation. Calling its vectors *state* highlights the separation between a shared operator and the runtime-selectable values supplied to it; it does not make those values cease to be parameters. The important empirical comparison is the restriction to an input-independent offset at each layer versus a learned input-dependent matrix update.

3.2 DOPAMINE: ONE ADDITIVE RESIDUAL VECTOR PER LAYER

For a pre-normalized decoder layer, let $H_\ell \in \mathbb{R}^{T\times d}$ be the residual stream over T positions. Writing the attention and MLP sublayers sequentially, the intervention is

$$A_\ell = H_\ell + \mathrm{Attn}_\ell(\mathrm{LN}^a_\ell(H_\ell)), \tag{2}$$

$$H_{\ell+1} = A_\ell + \mathrm{MLP}_\ell(\mathrm{LN}^m_\ell(A_\ell)) + \mathbf{1}_T b_{u,\ell}^\top, \qquad b_{u,\ell} \in \mathbb{R}^d. \tag{3}$$

Here LN denotes the architecture's normalization operator; the implementation adds the vector to the MLP output before residual addition. Each layer has its own vector, shared across positions for one user. The full user mode is $m_u = (b_{u,1}, \ldots, b_{u,L})$ and contains

$$P_{\mathrm{mode}} = \sum_{\ell=1}^{L} d_\ell = Ld \tag{4}$$

scalars when widths are constant. No learned projection, generator network, or shared personalization basis is required by this full-state variant.

We initialize every $b_{u,\ell}$ to zero. Before adaptation the intervention is an exact no-op, avoiding an initial change in the frozen model's function. We minimize masked next-token cross-entropy on the user's history,

$$\mathcal{L}_u(m_u) = -\frac{1}{Z_u} \sum_{(x,y)\in\mathcal{D}_u} \sum_{t\in\mathcal{T}_y} \log p_W(y_t \mid x, y_{<t}; m_u), \tag{5}$$

where $\mathcal{T}_y$ includes supervised target positions and Z_u counts them. Only the mode vectors are updated. Gradients must still propagate through the frozen network's computations to these vectors; freezing weights does not eliminate activation memory or backpropagation cost.

3.3 ROW-WISE COMPOSITION

As illustrated in Figure 1b, for a batch of K sequences with user IDs $u_1, \ldots, u_K$, gather the appropriate vectors into $M_\ell \in \mathbb{R}^{K\times 1\times d}$:

$$M_\ell[k, 1, :] = b_{u_k,\ell}, \qquad \widetilde{H}_{\ell+1} = H^{\mathrm{base}}_{\ell+1} + M_\ell. \tag{6}$$

Here $H^{\mathrm{base}}_{\ell+1}$ means the layer output before its additive intervention, not the activations of a separately run unconditioned model. All earlier layer outputs already depend on the selected mode. With the ordinary decoder assumption that attention does not cross batch rows, each row equals the corresponding single-user computation, up to numerical execution differences. The user vectors do not interact with one another. Appendix C.2 states the argument and gives pseudocode.

The extra arithmetic is one broadcast addition per intervened layer, $O(KTd)$ for that layer. This describes the intervention, not the whole forward pass. Loading and gathering profiles, memory traffic, batching policy, and kernel implementation can still dominate overhead; our main claim is a small per-user representation, not a measured end-to-end serving speedup.

3.4 FALSIFICATION PROBES

Two restricted alternatives ask what the representation must preserve. First, TAUGATE and MODECODE use multiplicative activation modulation, with direct or code-conditioned gate parameters. A gate rescales existing coordinates; an additive offset can inject a direction independently

of the current activation magnitude. This geometric distinction motivates the test but is not an explanation established by the experiments.

Second, concatenate a full additive mode into $b_u \in \mathbb{R}^{Ld}$ and restrict it to a fixed basis $B \in \mathbb{R}^{k \times Ld}$, learning only $c_u \in \mathbb{R}^k$:

$$b_u = B^\top c_u. \tag{7}$$

We refer to this restricted family as DOPAMINECODE. The basis may be random or obtained by PCA over learned user modes. Per-user storage falls to k scalars, but a shared basis must also be stored. These experiments test particular fixed-subspace parameterizations, not all possible learned encoders or compression objectives. For scale, a 64-dimensional fp16 basis for the 30,720-dimensional SmolLM2 mode would itself occupy about 3.75 MiB before any user codes. The reported compression result therefore concerns the tested per-user representation, not a complete system-level storage accounting.

4 EXPERIMENTAL SETUP

Backbones and baselines. We evaluate SmolLM2-360M, Qwen2.5-1.5B/3B/7B/14B, and Llama-3.2-1B/3B. LoRA-$r8$ is the common cross-model reference; the SmolLM2 rank sweep includes $r \in \{1, 2, 4, 8\}$. Parameter counts are the recorded trainable counts for each run, not an estimate based only on nominal model size. The synthetic controls include the unadapted backbone and multiplicative-state alternatives. We distinguish these controls from the broader set of related methods discussed in Section 2, which were not all run in this study.

Adaptation budget. The main protocol uses 60 Adam optimization steps and batch size 4. The activation-state learning rate is 10^{-2} and LoRA's is 5×10^{-4}, with LoRA scaling $\alpha = 2r$. Each method is adapted independently to each user's history. The fixed optimization budget makes the reported comparison concrete, but is not a guarantee that every baseline has reached its best attainable accuracy. We do not infer a training-time speedup from the difference in trainable parameter counts.

Controlled preferences. Synthetic users have deterministic preferences spanning verbosity, tone, format, casing, and sign-off. The principal SmolLM2 experiment uses a pool of 100 users, with 20 adaptation examples per user. Held-out teacher-forced cross-entropy is evaluated on 100 prompts per user; the generation-based style metric uses 16 held-out prompts per user. We call the archived rule-based generation metric *style accuracy*, and do not equate it with token accuracy or free-form human preference satisfaction. Additional seeds test sensitivity to the experimental draw. Each of the six non-SmolLM2 cross-model cells uses 30 users and one seed.

Amazon-review rating prediction. We construct a pool of 2,000 users with at least 24 reviews from the Books category of Amazon Reviews 2023 (Hou et al., 2024). The evaluated subsets are 300 users for SmolLM2-360M and 150 for Qwen2.5-1.5B, with 16 history reviews for adaptation and eight held-out reviews per user. The task predicts a 1–5 star rating from review text, using the user history to adapt the model. This is a LaMP-3-style construction, not the official LaMP evaluation (Salemi et al., 2024). A review can itself contain strong rating cues; the task does not isolate the contribution of latent user preferences from textual sentiment or user-specific label frequencies.

Metrics and aggregation. We report user-level generation accuracy averaged within each archived experimental cell, and held-out target-token cross-entropy (CE). Seed summaries average the reported seed means, rather than pooling all examples across seeds. Means and standard deviations across seeds describe observed variability; they are not confidence intervals. For ratings, the small eight-example test set gives a per-user accuracy resolution of 0.125. MAE is omitted because the archived implementation conditioned it on parseable outputs. Appendix A supplies additional protocol details and the aggregation conventions.

Table 1: **Synthetic style: parameter budget and accuracy.** Archived SmolLM2-360M seed-0 cells. Ratios divide each method's per-user parameter count by DOPAMINE's. Bold marks the best observed adapted score, not a significance claim.

Method	Per-user parameters	Relative state	Style accuracy ↑
Frozen	0	—	.3358
LoRA-$r1$	542,720	$17.7\times$	.8656
LoRA-$r2$	1,085,440	$35.3\times$	.8713
LoRA-$r4$	2,170,880	$70.7\times$	.8708
LoRA-$r8$	4,341,760	$141.3\times$	.8694
DOPAMINE	**30,720**	$1.0\times$	**.8811**

Table 2: **Three-seed controlled-preference results.** Values are per-seed means; the final row is the population standard deviation across seed means. The main comparison is Dopamine versus LoRA-$r8$, not a claim of statistical equivalence.

Seed	Frozen	DOPAMINE	LoRA-$r8$
0	.3358	.8811	.8694
1	.3044	.8458	.8508
2	.2702	.9021	.9079
Mean	.3034	.8763	.8760
Population SD	.0268	.0232	.0238

5 RESULTS

5.1 SMALL MODES RETAIN THE QUALITY OF MUCH LARGER WEIGHT UPDATES

Table 1 reports the SmolLM2 rank comparison. DOPAMINE reaches 0.881 style accuracy with 30,720 per-user parameters, compared with 0.866–0.871 for LoRA ranks 1–8 and 0.336 for the frozen backbone. Its observed accuracy exceeds each tested LoRA rank in these archived cells, while the parameter reduction ranges from $17.7\times$ against rank 1 to $141.3\times$ against rank 8. Increasing LoRA rank does not monotonically increase quality under this optimization protocol.

The three-seed comparison is more informative than a single-cell win. DOPAMINE obtains 0.8763 ± 0.0232 and LoRA-$r8$ obtains 0.8760 ± 0.0238, versus 0.3034 ± 0.0268 for the frozen model (population standard deviations over seed means). The direction of the DOPAMINE–LoRA gap changes across seeds: approximately $+0.0118$, -0.0050, and -0.0058. This supports *comparable observed quality*, not a consistent superiority claim. The seed means are retained separately in Table 2 and in the appendix; Figure 2 does not mix pooled seed means into the seed-0 rank plot.

5.2 THE PATTERN EXTENDS ACROSS SCALE AND MODEL FAMILIES

Table 3 compares the seven backbone/scale cells. The largest absolute gap between DOPAMINE and LoRA-$r8$ is approximately 0.012: DOPAMINE is ahead by this amount for SmolLM2-360M and behind by a similar amount for Llama-3.2-1B. Qwen2.5-1.5B, 3B, and 7B have equal reported means; Qwen2.5-14B and Llama-3.2-3B differ by less than 0.001 before rounding. Thus, no single family or larger model drives the result. The cross-family evidence is nevertheless limited by 30-user, single-seed cells outside SmolLM2.

The size reduction remains substantial as the backbone grows because the mode scales with residual width and layer count rather than the dimensions of all adapted matrices. Across the tested backbones, the recorded reduction against LoRA-$r8$ is 140–$215\times$. This range is a statement about the tested adapter configurations; changing LoRA's rank or target modules changes it. It is not a reduction in total model size, inference memory, or training compute.

Table 3: **Synthetic style across three families and seven scales.** SmolLM2 reports its seed-0 cell; the other backbones each use 30 users and one seed. State ratios compare LoRA-$r8$ with DOPAMINE. Counts in parentheses are DOPAMINE parameters, rounded to one decimal K.

Backbone	Frozen	DOPAMINE	LoRA-$r8$	State ratio
SmolLM2-360M (30.7K)	.3358	.8811	.8694	141.3×
Qwen2.5-1.5B (43.0K)	.2975	.8833	.8833	214.7×
Qwen2.5-3B (73.7K)	.2963	.8846	.8846	203.0×
Qwen2.5-7B (100.4K)	.2892	.8850	.8850	201.1×
Qwen2.5-14B (245.8K)	.2979	.8846	.8850	140.0×
Llama-3.2-1B (32.8K)	.2929	.8721	.8838	172.0×
Llama-3.2-3B (86.0K)	.2946	.8838	.8846	141.3×

Table 4: **Amazon rating prediction: all available main-task ranks.** SmolLM2 evaluates 300 users and Qwen2.5 evaluates 150, with eight held-out reviews per user. Bold indicates the best observed value within a backbone. Parameter counts exclude the shared backbone.

Backbone	Method	Accuracy ↑	CE ↓	Per-user params
SmolLM2-360M	Frozen	.5358	2.1870	0
	LoRA-$r1$	.7238	.7397	542.7K
	LoRA-$r2$	.7354	.7478	1.09M
	LoRA-$r4$	**.7392**	.7875	2.17M
	LoRA-$r8$	.7383	.8113	4.34M
	DOPAMINE	.7321	**.5575**	**30.7K**
Qwen2.5-1.5B	Frozen	.2217	8.2670	0
	LoRA-$r8$	.7125	1.0863	9.23M
	DOPAMINE	**.7275**	**.7513**	**43.0K**

5.3 AMAZON-REVIEW HISTORIES: SIMILAR ACCURACY WITH LOWER CROSS-ENTROPY

Table 4 reports the complete available SmolLM2 rating rank sweep. At 360M, DOPAMINE achieves 0.7321 accuracy, within 0.0071 of LoRA-$r4$ (the best tested rank) and 0.0063 of LoRA-$r8$. Its CE is 0.5575, below every tested LoRA rank (0.7397–0.8113). At Qwen2.5-1.5B, DOPAMINE reaches 0.7275 accuracy versus 0.7125 for LoRA-$r8$, with CE 0.7513 versus 1.0863. The corresponding state reductions against rank 8 are 141× and 215×.

Accuracy and CE answer different questions. Accuracy depends on the emitted rating; CE also responds to how probability is distributed over target tokens. The lower CE therefore adds evidence beyond the nearly tied discrete scores, but does not alone establish better calibration or better conversational personalization. In the 300-user SmolLM2 comparison, the paired per-user accuracy difference from LoRA-$r8$ is -0.00625 with standard deviation approximately 0.129, and 168 users have exactly tied accuracies. These summaries do not constitute a pre-specified equivalence test.

The rating result also has a task-specific caveat. Neither a per-user majority rating baseline nor a shuffled-history control is included in the retained comparison. Such controls are needed to separate user-specific adaptation from a learned response-format or label-prior correction. We therefore interpret this experiment as preserving the quality of a user-adapted LoRA baseline on an Amazon-review task, not as definitive evidence of richer user understanding.

5.4 NOT EVERY COMPACT ACTIVATION STATE IS A USEFUL MODE

Probability changes need not become useful generation changes. Multiplicative TAUGATE and MODECODE have generation accuracy near 0.3353, compared with 0.3358 for the unadapted model and 0.8811 for DOPAMINE. The archived gating sweeps explored gate strengths, layer subsets, code sizes, and adaptation budgets, but did not close this gap. Thus, reducing loss with a small activation intervention is not sufficient evidence that a useful personalized mode has been learned.

For DOPAMINE, a teacher-forced diagnostic reports mean greedy-token agreement with the frozen model of 0.48845, or about 51% changed top-1 predictions at the measured target positions. This

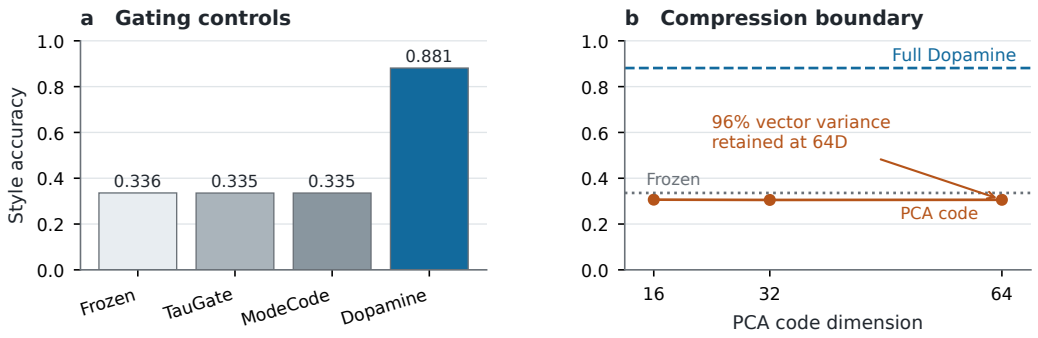


Figure 3: **Two failure boundaries.** (a) Archived multiplicative-state controls remain near frozen style accuracy, whereas the additive mode changes generation behavior substantially. (b) Shared PCA codes of 16, 32, or 64 dimensions fail to preserve the full mode's score. The reported 64-dimensional basis retains 96% of vector variance. This is a failure of the tested compression procedure, not a lower bound on intrinsic personalization dimension. Controls are from their respective archived runs; see Appendix B for scope.

is evidence that the intervention can affect token selection, rather than only confidence. It is not a free-generation token-edit rate. Comparable gate argmax diagnostics were not logged, so we do not infer that gating leaves all top-1 predictions unchanged or claim that this measurement establishes the cause of the performance gap.

Geometric compression can discard behaviorally important information. PCA codes of dimensions 16, 32, and 64 obtain style accuracies of 0.3065, 0.3052, and 0.3058, respectively, versus 0.3358 for the frozen model and 0.8811 for full DOPAMINE. A random 64-dimensional code is also unsuccessful (approximately 0.300). Capturing 96% of mode-vector variance does not guarantee preservation of the directions or optimization geometry needed for generation. The result does not prove that the remaining variance is causally necessary, that individual user modes are irreducibly high-dimensional, or that nonlinear, behavior-aware compression would fail. It establishes a practical boundary of the particular fixed-subspace test.

5.5 STORAGE AND SERVING: THE MEASURED CLAIM AND THE IMPLIED INTERFACE

In fp16, full DOPAMINE states occupy 60–480 KiB per user across the tested backbones. The corresponding LoRA-$r8$ states occupy 8.28–65.62 MiB, excluding the shared backbone in both cases. For SmolLM2, even LoRA-$r1$ uses about 1.04 MiB versus a 60 KiB mode. These are uncompressed tensor payloads, not serialized-file sizes; optimizer states, metadata, caches, and allocator overhead are excluded. Appendix C gives every exact parameter count.

The population-level accounting is linear in the number of stored users: with N profiles, the fp16 mode payload is $2NLd$ bytes. Thus the smallest and largest tested examples are 60 KiB versus 8.28 MiB per user at 360M, and 480 KiB versus 65.62 MiB at 14B, for DOPAMINE and LoRA-$r8$ respectively. This is a storage comparison, not a claim about accelerator residency or end-to-end serving cost.

The mode interface supports row-wise profile selection (Section 3.3) without per-user matrix materialization. This is a useful architectural consequence of the representation, but the archived multi-user timing study used the earlier TAUGATE controller, not DOPAMINE. We retain those measurements only as a separately labeled historical systems experiment in Appendix C.3. We do not attribute its 16-user batching speedup to DOPAMINE, compare DOPAMINE to optimized S-LoRA or Punica without measurements, or claim that LoRA must be served sequentially.

As a supplementary task check, we also evaluated held-out Amazon-review text likelihood rather than rating-token prediction. On SmolLM2, target CE was 3.5208 for the frozen model and 3.2746

with DOPAMINE; LoRA-$r8$ reached 5.4137 under the same fixed optimization budget. We treat this as a task-specific observation, not evidence that LoRA cannot personalize prose: the fixed settings may be poorly suited to that comparison, and we have no open-ended human evaluation or capability-retention test. It nevertheless shows that the activation interface is not restricted to selecting discrete rating labels. Full details are in Appendix B.

6 DISCUSSION AND LIMITATIONS

What the evidence changes. The study challenges a default choice of representation, not the usefulness of weight adaptation. On the tested tasks, a user-specific constant offset per layer retains much of the behavior obtained by millions of LoRA parameters. One interpretation is that the shared model already supplies much of the needed computational repertoire and the learned vectors bias which trajectories are expressed. The experiments establish functional sufficiency under the tested protocol, not this mechanistic explanation.

Where the conclusion may not transfer. Synthetic style rules and review ratings cover a narrow portion of personalization. We do not test persistent conversational memory, factual knowledge unique to a user, preference drift, conflicting preferences, multilingual behavior, or human judgments of personalized helpfulness. Each mode is learned separately; cold-start generalization to unseen users and amortized history-to-mode inference are not evaluated. No claim is made that a mode learned for one backbone transfers to another.

Comparison and uncertainty. Cross-model cells are small and mostly single-seed. Fixed learning rates and training budgets do not establish an optimally tuned PEFT frontier. The main comparison is LoRA, not a comprehensive study of prompt tuning, prefix tuning, ReFT, BiPO, or bias-only alternatives. *LoRA-level* means comparable observed scores in the reported cells; it is neither a statistical equivalence guarantee nor a claim of universal dominance. The Amazon-review task requires stronger history and label-prior controls to support a more specific claim about individual preference learning.

State is still an adaptation artifact. A small vector is not intrinsically safe, interpretable, or private. It can encode sensitive behavioral information and can alter model outputs in undesired ways. We do not measure privacy leakage, capability retention on unrelated tasks, adversarial robustness, or interactions with safety training. A deployment would need user consent, access controls, profile isolation, deletion mechanisms, and evaluations appropriate to its use case. Reducing the state size does not remove these obligations.

7 CONCLUSION

We may be representing personalization at the wrong level. DOPAMINE operationalizes a different choice: retain one frozen set of model weights and represent each user by a small, learned activation mode. Across the evaluated families, scales, and tasks, this restricted interface achieves quality comparable to LoRA with 140–215$\times$ fewer per-user parameters than rank 8. Gating and compression failures delimit the result, while row-wise composition exposes a natural many-user interface. The lesson is not that users never need adapted models. It is that *a different effective model for every user should be a hypothesis to justify, rather than the starting assumption*.

REPRODUCIBILITY STATEMENT

Section 3 specifies the intervention and objective; Section 4 and Appendix A describe the experimental protocol and aggregation. Tables 1–4 and Appendices B–C retain the quantitative comparisons, seed summaries, and state sizes. Figure-generation source and the aggregate values underlying the plots accompany the manuscript source. These aggregates reproduce the figures, not the original training runs.

ETHICS STATEMENT

The Amazon-review study uses public review data and reports aggregate results, not reviewer identities or individual profiles. Public availability does not remove privacy concerns. Learned modes should be treated as user-profile data, not as anonymous merely because they are compact. The work does not establish that personalized models are safe to deploy, nor does it evaluate persuasive manipulation, discrimination, or sensitive attribute inference. The synthetic preferences are controlled behavioral rules and should not be interpreted as a model of human identity.

AI USE STATEMENT

Generative AI tools assisted with the conceptual framing, interpretation and audit of reported results, literature discovery, manuscript drafting and editing, and figure and analysis-code preparation. These tools were used to reconstruct the manuscript from retained text and experimental summaries; this editing process did not conduct new training experiments. Responsibility for verifying the experimental record, citations, and final submission remains with the authors.

REFERENCES

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2022. URL https://arxiv.org/abs/2106.10199.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.00045.

Lequn Chen, Zihao Ye, Yongji Wu, Danyang Zhuo, Luis Ceze, and Arvind Krishnamurthy. Punica: Multi-tenant LoRA serving. *arXiv preprint arXiv:2310.18547*, 2023. URL https://arxiv.org/abs/2310.18547.

Sumanth Doddapaneni, Krishna Sayana, Ambarish Jash, Sukhdeep Sodhi, and Dima Kuzmin. User embedding model for personalized language prompting. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems*, pp. 124–131, 2024. doi: 10.18653/v1/2024.personalize-1.12. URL https://aclanthology.org/2024.personalize-1.12/.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952v1*, 2024. URL https://arxiv.org/abs/2403.03952v1.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, 2019. URL https://arxiv.org/abs/1902.00751.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021. URL https://arxiv.org/abs/2104.08691.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021. URL https://arxiv.org/abs/2101.00190.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2205.05638.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024. URL https://arxiv.org/abs/2304.11406.

Ying Sheng, Shiyi Cao, Dacheng Li, Coleman Hooper, Nicholas Lee, Shuo Yang, Christopher Chou, Banghua Zhu, Lianmin Zheng, Kurt Keutzer, Joseph E. Gonzalez, and Ion Stoica. S-LoRA: Serving thousands of concurrent LoRA adapters. In *Proceedings of Machine Learning and Systems*, 2024. URL https://arxiv.org/abs/2311.03285.

Viacheslav Sinii, Alexey Gorbatovski, Artem Cherepanov, Boris Shaposhnikov, Nikita Balagansky, and Daniil Gavrilov. Steering LLM reasoning through bias-only adaptation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9202–9211, 2025. doi: 10.18653/v1/2025.emnlp-main.467. URL https://aclanthology.org/2025.emnlp-main.467/.

Nishant Subramani, Nivedita Suresh, and Matthew E. Peters. Extracting latent steering vectors from pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2022. URL https://arxiv.org/abs/2205.05124.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.15213.

Alex Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248v1*, 2023. URL https://arxiv.org/abs/2308.10248v1.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2404.03592.

A PROTOCOL AND AGGREGATION DETAILS

Scope of the experimental record. The main comparisons use separately trained per-user interventions with a fixed backbone. The synthetic and Amazon experiments are distinct tasks, and the multiplicative-control and serving experiments are separate archived studies. Their numbers should not be pooled into a single matched experiment. Table 5 summarizes the retained protocol details.

Table 5: Adaptation and evaluation settings used in the main comparisons.

Setting	Value
Updated backbone weights	None
Full mode intervention	Additive MLP-output vector at every layer
Mode initialization	All zeros; exact no-op
Mode parameter count	Ld_{model}
Optimizer and steps	Adam; 60 steps
Batch size	4
Mode learning rate	10^{-2}
LoRA learning rate and scaling	5×10^{-4}; $\alpha = 2r$
SmolLM2 LoRA ranks	1, 2, 4, 8
Other-model LoRA reference	Rank 8
Synthetic adaptation set	20 examples per user
Synthetic evaluation	100 CE prompts; 16 generation prompts per user
Cross-model synthetic cells	30 users per non-SmolLM2 cell; one seed
Amazon candidate pool	2,000 Books users with at least 24 reviews
Amazon evaluated users	300 at 360M; 150 at 1.5B
Amazon split per user	16 adaptation reviews; 8 held-out reviews

Synthetic task. The five preference dimensions define a controlled behavioral target, not a distribution of real human personalities. A user history provides examples of that target, and generation is scored for compliance with the synthetic preferences. Teacher-forced CE uses target tokens from a larger held-out set than the generation metric. These metrics must not be conflated: improved probability on a reference continuation may leave the decoded continuation unchanged, and a decoded style score does not quantify general answer quality. The retained aggregate metric is described as style accuracy throughout; we do not reinterpret it as strict whole-response exact match.

Amazon preprocessing. The preparation procedure groups Books reviews by reviewer identifier and retains a candidate pool of users reaching the minimum 24-review history. The retained preprocessing summary truncates review text to 600 characters and titles to 120 characters. Sixteen reviews supply adaptation examples and eight supply evaluation examples for each evaluated user. This is a fixed-history prediction setting: the paper does not claim chronological forecasting or evaluation of evolving preferences. The pool size of 2,000 is not the number of evaluated users. No review text or reviewer identifier is included in the figure-data package.

Statistical unit. Let $a_{u,s}$ denote a user's recorded accuracy in seed s, and let $\mathcal{U}_s$ be the evaluated users for that cell. We report

$$\bar{a}_s = \frac{1}{|\mathcal{U}_s|} \sum_{u \in \mathcal{U}_s} a_{u,s}, \qquad \bar{a} = \frac{1}{S} \sum_{s=1}^{S} \bar{a}_s, \qquad \sigma_{\text{seed}} = \sqrt{\frac{1}{S} \sum_{s=1}^{S} (\bar{a}_s - \bar{a})^2}. \tag{8}$$

The last quantity is the population standard deviation of the available seed means, not an estimate of the standard error across individual prompts. In particular, averaging seed means does not require, or establish, identical user counts or identical users across seeds. The retained summaries are not sufficient to construct a new paired confidence interval across the three seeds.

Paired rating comparison. For the 300-user SmolLM2 rating experiment, define $d_u = a_u^{\text{Dopamine}} - a_u^{\text{LoRA-}r8}$. The retained mean is -0.00625 and the per-user standard deviation is approximately 0.129; 168 of 300 users have $d_u = 0$. With eight held-out examples per user, ties are expected even when predicted probabilities differ. These statistics are compatible with both methods

Table 6: SmolLM2 synthetic seed means. All columns use the same seed-summary convention; no cross-seed pooling is used in the rank or scale figures. Values are shown to six decimals to expose the small aggregate difference.

Seed	Frozen	DOPAMINE	LoRA-$r8$
0	.335750	.881125	.869375
1	.304375	.845833	.850833
2	.270208	.902083	.907917
Mean	.303444	.876347	.876042
Population SD	.0268	.0232	.0238

being useful under the protocol but do not demonstrate pre-specified statistical equivalence. They also do not identify whether any difference is concentrated among users with particular histories.

B NEGATIVE RESULTS AND SUPPLEMENTARY TASK

B.1 MULTIPLICATIVE-STATE CONTROLS

Table 7 preserves the generation scores from the archived multiplicative-control study and the full additive reference. The LoRA score in that older study was 0.8881, from a different run than the 0.8694 seed-0 rank-sweep reference. We do not substitute the former into the rank or scale plots. The distinction matters because otherwise an apparent method gap would partly reflect a change in run selection.

Table 7: Synthetic style controls. Values from separate archived study groups are identified explicitly and are not treated as a matched-seed ablation.

Study group	Method	Style accuracy
Multiplicative controls	Frozen	.33575
	TAUGATE	.3353
	MODECODE	.3353
	IA^3	.3354
	Prompt-only control	.3387
	LoRA (older control run)	.8881
Additive study, seed 0	DOPAMINE	.881125

The gate search varied strength values $\{0.03, 0.1, 0.3\}$, layer selection (last quarter, middle half, or all layers), gain-scale values $\{1, 3, 10\}$, code sizes from 16 to 256, and training duration. These are reported search axes, not a claim that every possible Cartesian combination was evaluated or that every multiplicative intervention must fail. The result is specific to the tested controls and their optimization. The gate study shows why loss reduction alone is an inadequate success criterion for a generation-based personalization claim.

B.2 ARGMAX DIAGNOSTIC

The additive diagnostic compares the most probable token with and without the learned intervention under teacher forcing:

$$q_u = \frac{1}{|\mathcal{T}_u|} \sum_{t \in \mathcal{T}_u} \mathbf{1}\Big[\arg\max_v p_W(v \mid x, y_{<t}; m_u) = \arg\max_v p_W(v \mid x, y_{<t}; 0)\Big]. \tag{9}$$

The reported mean agreement is 0.48845; its complement is 0.51155. The contexts on both sides are supplied reference prefixes, so this statistic must not be interpreted as an alignment or edit distance between two independently generated sequences. The archive does not provide the same diagnostic for the gate controls; missing values are not zero changes.

B.3 FIXED-SUBSPACE COMPRESSION

Table 8: Compression of the additive user mode. The full mode has 30,720 parameters per user on SmolLM2. A code also requires a shared basis. The 96% variance figure is reported for the 64-component PCA basis constructed from 100 learned modes.

Representation	Per-user code size	Style accuracy
Frozen backbone	0	.33575
Full DOPAMINE	30,720	.881125
PCA DOPAMINECODE	16	.30646
PCA DOPAMINECODE	32	.30521
PCA DOPAMINECODE	64	.30583
Random-basis DOPAMINECODE	64	$\approx$ .300

PCA optimizes reconstruction of the vector distribution in Euclidean space, not preservation of conditional output distributions. With a fixed basis, optimizing a user's code also constrains every adaptation update to the basis span. The observed degradation could arise from discarded directions, the optimization problem in code space, the construction of the basis, or a combination. Distinguishing these explanations would require, for example, directly projecting already-trained modes and comparing that operation with re-optimizing codes, or training a basis with a behavioral objective. Those follow-up experiments are not part of the reported evidence.

B.4 AMAZON REVIEW LANGUAGE MODELING

A supplementary SmolLM2 experiment evaluates held-out review language modeling for 300 users, rather than prediction of a rating token. The recorded CE is 3.5208 for the frozen model, 3.2746 for DOPAMINE, and 5.4137 for LoRA-$r8$. These results retain an additional observation from the original experiment summaries, but the unfavorable LoRA result should not be interpreted as an intrinsic inability to personalize prose: the fixed optimization settings may be poorly suited to this task. No open-ended human evaluation, broad hyperparameter sweep, or general capability-retention study is supplied for this comparison.

C EXACT STATE SIZES AND ROW-WISE COMPOSITION

C.1 TENSOR PAYLOAD ACCOUNTING

Table 9 uses two bytes per parameter for both methods: $\mathrm{KiB} = 2P/1024$ and $\mathrm{MiB} = 2P/1024^2$. The ratio is unchanged by any common scalar precision. A compressed file, training checkpoint, or deployment process will have a different footprint because it may contain metadata, optimizer moments, gradients, padding, and other state.

Table 9: Exact recorded per-user parameter counts and fp16 tensor payloads. The shared backbone is excluded from both methods.

Backbone	Mode params	Mode KiB	LoRA-$r8$ params	LoRA MiB
SmolLM2-360M	30,720	60	4,341,760	8.28
Qwen2.5-1.5B	43,008	84	9,232,384	17.61
Qwen2.5-3B	73,728	144	14,966,784	28.55
Qwen2.5-7B	100,352	196	20,185,088	38.50
Qwen2.5-14B	245,760	480	34,406,400	65.62
Llama-3.2-1B	32,768	64	5,636,096	10.75
Llama-3.2-3B	86,016	168	12,156,928	23.19

For a population of N stored users, the additive payload scales as $2NLd$ bytes in fp16. This is population storage, not the amount that must be resident on an accelerator at once. Likewise, a LoRA server can keep adapters outside accelerator memory and load or batch them as needed. The

representation changes the size and operator associated with an individual profile; serving policy remains an independent systems choice.

C.2 Composition property and pseudocode

Property. Assume each transformer operation acts independently on batch rows and uses the same token mask, positions, and mode as the corresponding single-user execution. Then evaluating a batch with row-selected additive modes gives, for each row, the same mathematical output as evaluating that row alone with its mode.

Argument. At the input, each batch row contains the same token embeddings as its single-user counterpart. Suppose the residual streams agree at layer ℓ. The shared attention and MLP operations use no other batch row, so their outputs agree. Both executions add the same $b_{u,\ell}$ to every position. Their residual streams therefore agree at layer $\ell + 1$. Induction through all layers, followed by the shared output head, proves the statement. Different kernel schedules may introduce ordinary floating-point differences.

Conceptual mixed-user forward (not a benchmark implementation)
$H \leftarrow \mathrm{Embed}(X)$
for layer $\ell = 1, \ldots, L$:
  $A \leftarrow H + \mathrm{Attn}_\ell(\mathrm{LN}^a_\ell(H))$
  $V \leftarrow \mathrm{MLP}_\ell(\mathrm{LN}^m_\ell(A))$
  $M \leftarrow \mathrm{Gather}(\text{mode table}_\ell, \text{user IDs}) \quad [K, d]$
  $H \leftarrow A + V + M[:, \mathrm{None}, :] \quad [K, T, d]$
return $\mathrm{LMHead}(\mathrm{FinalNorm}(H))$

This property assumes a user's mode is held fixed for the sequence. Switching modes partway through cached decoding without recomputing the prefix would retain keys and values produced under the previous mode. Consequently, swapping a profile and reusing an old cache is not generally equivalent to a fresh evaluation under the new profile. Row-wise composition should not be conflated with arbitrary mid-sequence mode switching.

C.3 Historical gate-serving measurements

For completeness, Table 10 retains the archived systems measurements. **The activation controllers in this benchmark are TauGate and ModeCode, not Dopamine.** These measurements are not used to establish the main paper's quality or speed claims.

Table 10: Historical serving study, reported separately from Dopamine. Throughput depends on the archived implementation and workload.

Single-user variant	Tokens/s
Frozen backbone	48.48
TauGate	46.90
ModeCode	45.70
IA^3	41.08
Unmerged LoRA	27.98
Merged LoRA	46.99

In the archived 16-user comparison, batched TauGate took 1.1985 seconds (640.8 tokens/s) and sequential merged-LoRA evaluation took 16.232 seconds (47.3 tokens/s), a ratio of about 13.54. This comparison changes batching as well as the adaptation operator. It is therefore neither a measurement of Dopamine nor evidence that optimized multi-LoRA serving is inherently 13.54× slower. A direct serving study should hold hardware, generated token counts, batching, caching, and memory accounting constant and include optimized many-adapter implementations.

D CLAIM-TO-EVIDENCE MAP

Table 11: What each headline claim does, and does not, establish.

Claim	Supporting comparison	Boundary
Small activation modes retain LoRA-level quality	Tables 1, 3, 4, 6	Observed protocol-specific means; no universal superiority or equivalence test.
140–215$\times$ less per-user state	Recorded counts in Table 9	Relative to LoRA-$r8$, not total model memory or compute.
Real histories support the pattern	300-user and 150-user rating cells	Not official LaMP; label-prior and shuffled-history controls absent.
Not every activation state suffices	Table 7 and Figure 3	Tested gate families only; no gate argmax diagnostic.
PCA variance need not preserve behavior	Table 8	Not a proof of an intrinsic dimensionality lower bound.
Modes compose row-wise	Equation 6 and Appendix C.2	Algebraic property; no direct Dopamine serving benchmark.
Brain modes, not brains	Functional metaphor for the personalization interface	No biological, cognitive, conscious, or anthropomorphic claim.