Steerable but Not Decodable: Function Vectors Operate Beyond the Logit Lens

Anonymous Author(s)
Affiliation
Address
email

Abstract

Function vectors (FVs), mean-difference directions extracted from in-context demonstrations, can steer language model behavior when added to the residual stream. We hypothesized that FV steering failures would coincide with information absence, so the parameter-free logit lens would fail wherever steering fails. We were wrong. Across 12 tasks, 6 models, and 4,032 directed cross-template pairs, we find the opposite dissociation: FV steering routinely succeeds where the logit lens cannot decode the correct answer at any intermediate layer, while the predicted converse pattern (decodable without steerable) is nearly empty (3/72). The dissociation survives a tuned-lens dialect-correction check (93% of gaps persist) and a 2-layer MLP probe with a Hewitt & Liang control (5/10 SAND cells close as nonlinearly encoded; 5 remain invisible to every decoder we tried). FVs achieving >0.90 steering accuracy still project to incoherent token distributions, indicating they encode computational instructions rather than answer directions. Activation patching causally confirms the difficulty hierarchy, and post-steering analysis exposes a model-family divergence: Mistral FVs rewrite intermediate representations while Llama and Gemma FVs do not, despite equally successful steering. The previously reported negative cosine–transfer correlation dissolves at scale. These findings challenge the assumption that additive steering operates through linearly decodable subspaces, with direct implications for safety monitoring of steering interventions.

1 Introduction

Activation steering, adding a learned direction to a language model's hidden states at inference time, is an emerging tool for controlling LLM behavior without retraining [3, 18, 41]. The approach rests on the *linear representation hypothesis* (LRH) [27]: task-relevant behaviors correspond to linear directions in activation space, so a function vector (FV) [14, 35] extracted from in-context learning (ICL) demonstrations should reliably steer the model when added to its hidden states.

For safety-critical deployment the relevant test is not whether an FV works on the same prompt template from which it was extracted, but whether it transfers to different template formulations of the same task [29, 40]. Prior FV work evaluates almost exclusively in-distribution (IID) [14, 19, 35]; a preliminary study [1] reported a negative cosine–transfer correlation across 3 tasks on a single 8B base model. We test whether this and related claims survive at scale across 12 tasks, 6 models from 3 families, and task-specific templates with no shared surface forms across tasks.

We frame four research questions: (1) for which task types does mean-difference FV extraction succeed; (2) is cross-template transfer predicted by geometric alignment or by IID efficacy; (3) when FV steering succeeds, can the model's own logit lens decode the correct answer at intermediate layers; and (4) how do these patterns vary across model families and instruction tuning. Combining FV

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

steering with the logit lens, we construct a 2×2 matrix: {logit lens decodes, fails} $\times$ {FV steering succeeds, fails}. We expected the upper-right cell, decodability without steerability, to be populated. The data invert this prediction: that cell contains only 3 of 72 task$\times$model instances (all Mistral), while the opposite cell, steerability without decodability, contains 10 spanning all families except Gemma Base. Unlike linear probing, the logit lens uses no learned parameters, sidestepping the probe-complexity critique [16].

Contributions. (1) **The largest cross-template FV transfer study to date**: 12 tasks, 6 models, 4,032 directed pairs, with pre-registered difficulty predictions. (2) **Systematic quantification of the steerability-without-decodability dissociation**, with the predicted opposite cell nearly empty (3/72). Related individual observations have been reported [2, 20, 30, 32, 35, 38]; our contribution is the multi-model, multi-task quantification with tuned-lens and MLP-probe robustness checks. (3) **Decoder-ladder robustness**: per-layer diagonal tuned-lens correction closes 1 of 14 SAND gaps (93% persist); a 2-layer MLP probe with Hewitt & Liang selectivity closes 5 of 10 cells while leaving 5 invisible to every decoder we tried. (4) **Model-family mechanistic divergence**: Mistral FVs rewrite intermediate representations (delta up to $+0.91$); Llama and Gemma FVs produce near-zero changes despite equally successful steering. Activation patching corroborates the difficulty hierarchy. The previously reported negative cosine–transfer correlation dissolves at scale ($|r| \leq 0.20$, $\Delta R^2 \leq 0.011$).

2 Background and Related Work

Function vectors and steering. ICL creates task-specific directions in activation space [14], formalized as function vectors via causal mediation [35], with reliability varying across model families [10]. Activation steering by mean-of-differences [25, 28, 36] is theoretically and empirically strong among extraction methods [17]. Recent work has questioned the reliability of steering vectors: Tan et al. [33] report substantial per-sample anti-steerability; Braun et al. [6] find directionally distinct yet behaviorally comparable vectors from prompt variants; Goldman-Wetzler & Turner [13] found $>$800 nearly orthogonal "write code" vectors all inducing the same behavior.

The linear representation hypothesis. Park et al. [26, 27] distinguish an unembedding representation (probing) from an embedding representation (steering) and unify them via a causal inner product. We provide empirical evidence that these come apart in the opposite direction from naive intuition.

Probing, logit lens, and prior dissociation observations. The logit lens [23] uses no learned parameters, sidestepping probe-complexity concerns [4, 16]; the tuned lens adds per-layer affine corrections [5]. Closely related dissociation observations include unreachable backtracking logits at early-mid layers [38], SAE input/output features rarely co-occurring [2], decodability-before-steerability during training [30], and steering via dormant pathways [20]. Our contribution is the systematic multi-model, multi-task quantification with a tuned-lens robustness check. Patching methodology follows causal tracing [8, 22].

3 Experimental Setup

Task battery. We design 12 tasks spanning 5 computational categories: Lexical Retrieval (`antonym`, `synonym`, `hypernym`), Factual Retrieval (`country_capital`, `english_spanish`, `object_color`), Morphological Transform (`past_tense`, `plural`), Character/Surface (`capitalize`, `first_letter`, `reverse_word`; predicted to fail), and Compositional/Semantic (`sentiment_flip`; predicted to fail). Pre-registered difficulty predictions, role assignments, and per-task input–output counts ($n \in [60, 95]$) are in Appendix K; the full taxonomy and motivation are in Appendix A.

Templates and models. Each task has 8 task-specific templates across 4 style categories (Natural, Symbolic, Question, Formal); no two tasks share any template string, and symbolic templates use unique function names (eliminating the shared-template confound of prior work). All 96 templates are listed in Appendix A. Eight templates yield 56 directed cross-template pairs per task, 4,032 total per model. We evaluate 6 models from 3 families, size-matched at 7–9B: Llama-3.1-8B (32 layers,

d=4096), Gemma-2-9B (42 layers, d=3584), and Mistral-7B-v0.3 (32 layers, d=4096), each in Base and Instruct variants.

FV extraction and steering. We extract FVs via the standard mean-of-differences method [17, 25, 28, 35]. For task t, template k, layer ℓ:

$$\mathrm{FV}_{t,k,\ell} \;=\; \mathbb{E}\big[\mathbf{h}_\ell^{(\mathrm{pos})}\big] - \mathbb{E}\big[\mathbf{h}_\ell^{(\mathrm{neg})}\big], \qquad \mathbf{h}'_\ell \;=\; \mathbf{h}_\ell + \alpha \cdot \mathrm{FV}_{t,k,\ell}, \tag{1}$$

with positive inputs being 15 correct ICL demonstrations, negative inputs being 15 zero-shot prompts; we sweep $\alpha \in \{0.5, \ldots, 5.0\}$ with adaptive refinement and report best accuracy across α and layers. IID evaluation uses the same template for extraction and test; OOD uses different templates. A task on a model passes the IID gate if mean IID accuracy exceeds $\tau = 0.10$; tasks failing the gate are excluded from causal analyses.

Logit lens, FV vocabulary projection, and tuned lens. For each layer ℓ, we project the zero-shot residual stream and each FV through the model's final LayerNorm and unembedding W_U [23] and report top-k accuracy ($k \in \{1, 5, 10\}$) and the FV *correct-output fraction* (proportion of top-50 tokens matching any correct task output). For `sentiment_flip` we substitute polarity classification (Appendix Q). To rule out representational dialect, we additionally apply the tuned lens [5] via per-layer diagonal affine translators $T_\ell(\mathbf{h}) = (\mathbf{1} + \boldsymbol{\delta}_\ell) \odot \mathbf{h} + \mathbf{b}_\ell$, identity-initialized and MSE-trained against the final-layer activation. The diagonal constraint is deliberately conservative: it captures per-feature scale and offset (the dominant source of dialect) without learned classification capacity. Training details are in Appendix L.

The steerability–decodability matrix. We classify each task$\times$model instance by whether FV steering accuracy and best-layer logit-lens top-10 each exceed $\tau = 0.10$, yielding the 2×2 matrix {decode succeeds, fails} $\times$ {steer succeeds, fails} (Figure 2). Naive intuition predicts the upper-right cell (decodable without steerable) would be populated; our data invert this. Baselines are zero-shot and 5-shot ICL. Geometric analysis: pairwise cosine between template FVs, alignment–transfer Pearson r, and hierarchical regression $\mathrm{OOD} \sim \mathrm{task} + \mathrm{cosine}$ with incremental ΔR^2.

4 Steering Efficacy and Cross-Template Transfer

IID steering. Figure 1 reports IID accuracy. Pre-registered predictions were exceeded for morphological tasks (`past_tense` 0.82–0.98, `plural` 0.85–1.00) and falsified for two of three character "negative controls" (`capitalize` 0.41–0.87 on Llama and Gemma, near-zero on Mistral; `first_letter` 0.36–0.78 across all 6 models; only `reverse_word` behaved as predicted). Zero-shot baseline accuracy correlates strongly with IID steering ($r = 0.72$–0.87; Appendix B), and instruction tuning consistently improves steering (mean Δ: $+0.21$ Llama, $+0.09$ Gemma, $+0.07$ Mistral; per-task in Appendix J). Aggregate destructive cases (mean steered $<$ zero-shot) occur at 0–1% across all models, compatible with prior per-sample anti-steerability reports [6, 33].

Cross-template transfer. For tasks passing the IID gate, transfer barely degrades: top tasks (`plural`, `antonym`, `past_tense`) show IID–OOD gaps below 0.10; the largest gap is `english_spanish` (-0.176 on Llama Base), driven by a few weak source templates. Source-template IID quality strongly predicts OOD transfer ($r = 0.80$–0.93 across all 6 models), so cross-template "failure" is primarily a source-quality problem (per-model tables: Appendix E). Template surface form does not affect transfer: within-style and across-style means are statistically indistinguishable on every model ($p > 0.30$; Appendix D). FVs capture an abstract style-invariant instruction.

Geometric analysis: the cosine–transfer correlation dissolves at scale. The pooled Pearson correlation between cosine and OOD transfer ranges from $r = -0.199$ (Mistral Base) to $r = +0.126$ (Gemma IT); within-task correlations scatter around zero. Hierarchical regression confirms (Table 1): task identity explains 40–58% of variance, and cosine adds at most $\Delta R^2 = 0.011$. Mistral's negative pooled r is a textbook Simpson's paradox: easy tasks cluster at high alignment and high transfer, hard tasks scatter at varied alignment and low transfer, tilting the pool. Cosine alignment tells you essentially nothing about transfer once you know the task. This complements reports that steering-vector geometry is not a reliable behavioral proxy [6, 13, 33]. Per-task dissociation rates and permutation tests are in Appendices F–G.

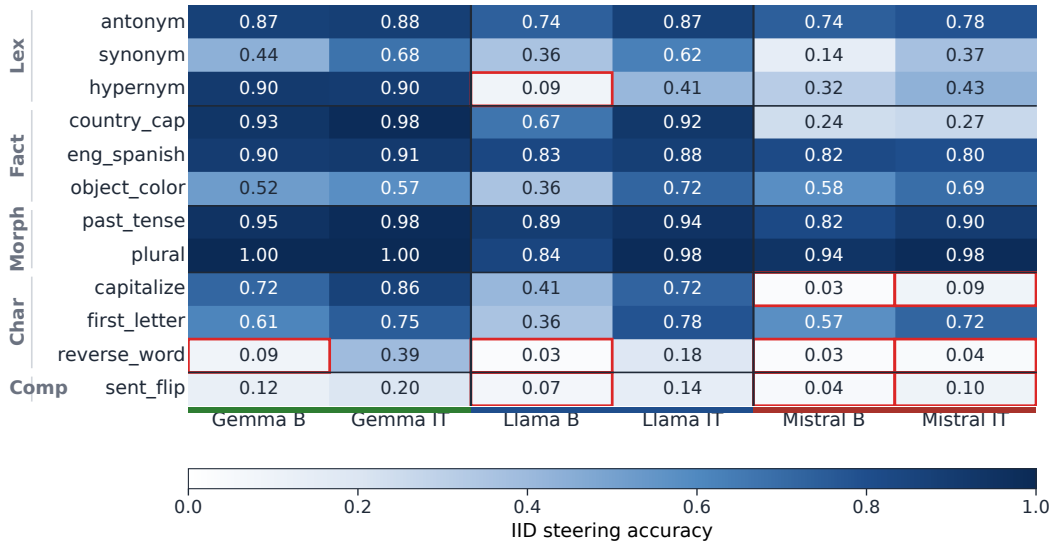

Figure 1: IID steering accuracy across 12 tasks and 6 models (mean across 8 templates at best layer and best α). Red borders mark cells failing the IID gate ($\tau = 0.10$); orange stars mark falsified negative-control predictions. `hypernym` shows the largest cross-model divergence (0.085 Llama Base vs. 0.902 Gemma Base), indicating task category predicts FV success only as a proxy for model competence. Per-family baselines: Appendix B.

Table 1: Hierarchical regression: task identity vs. cosine as predictors of OOD transfer. Cosine contributes at most 1.1 percentage points of R^2 beyond task identity on every model.

Model	R^2_{task}	$R^2_{\text{task+cos}}$	ΔR^2	Cos. coef
Gemma-2 Base	0.481	0.481	<0.001	−0.002
Gemma-2 IT	0.528	0.530	0.003	+0.131
Llama-3.1 Base	0.402	0.410	0.009	+0.153
Llama-3.1 IT	0.512	0.523	0.011	+0.235
Mistral Base	0.527	0.533	0.006	+0.191
Mistral IT	0.585	0.586	0.002	+0.118

5 The Steerability–Decodability Inversion

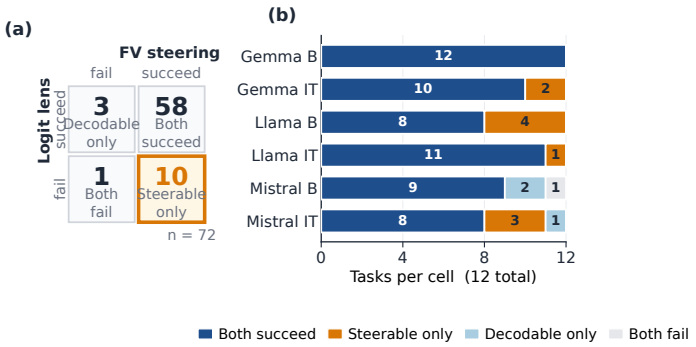

Figure 2: The populated 2×2 steerability–decodability matrix across all 72 task$\times$model instances. Circle area reflects count. The predicted upper-right cell (decodable without steerable) is nearly empty (3/72, all on Mistral); the lower-left cell (steerable without decodable) contains 10. The most dramatic case is `country_capital` on Llama Base: FV steering reaches 0.880 at L4 while logit-lens top-10 readability is 0.056 at L32. Per-model breakdown: Appendix L.

5.1 Logit-Lens Decodability vs. FV Steering

Across 72 task×model instances (Figure 2), the upper-right cell (decodable without steerable) contains only 3 instances, all on Mistral and corresponding to tasks where pre-training appears to have already encoded the answer in unembedding-aligned form. The lower-left cell (steerable without decodable) contains 10 instances spanning Llama Base (`country_capital`, `first_letter`, `object_color`, `hypernym`), Llama IT (`first_letter`), Gemma IT (`first_letter`, `object_color`), and Mistral IT (`country_capital`, `object_color`, `synonym`). The gap is negative for every (task, model) pair, with extremes of -0.91 (`first_letter` on Llama IT) and -0.82 (`country_capital` on Llama Base, the most dramatic case: FV at L4 gives 0.880 while peak logit-lens readability is 0.056 at L32). Per-task comparison: Appendix P.

FV vocabulary projection confirms the instruction interpretation. Even at >0.90 steering accuracy (`past_tense`, `plural`, `english_spanish`), projecting the FV through W_U produces incoherent token distributions: top-50 tokens include `deo`, `aber`, `aday`, `oman`, with correct-output fractions near zero (mean 0.003–0.035 across models). The partial exception is `first_letter`, whose FV projects to single-character tokens (∼0.82 task-relevant fraction on Llama Base) but not to the correct specific letters: it captures the "character extraction" operation without encoding which character. This replicates and extends the six-task observation of Todd et al. [35]: FVs encode computational instructions, not answer vectors.

5.2 Layer-Wise Profiles

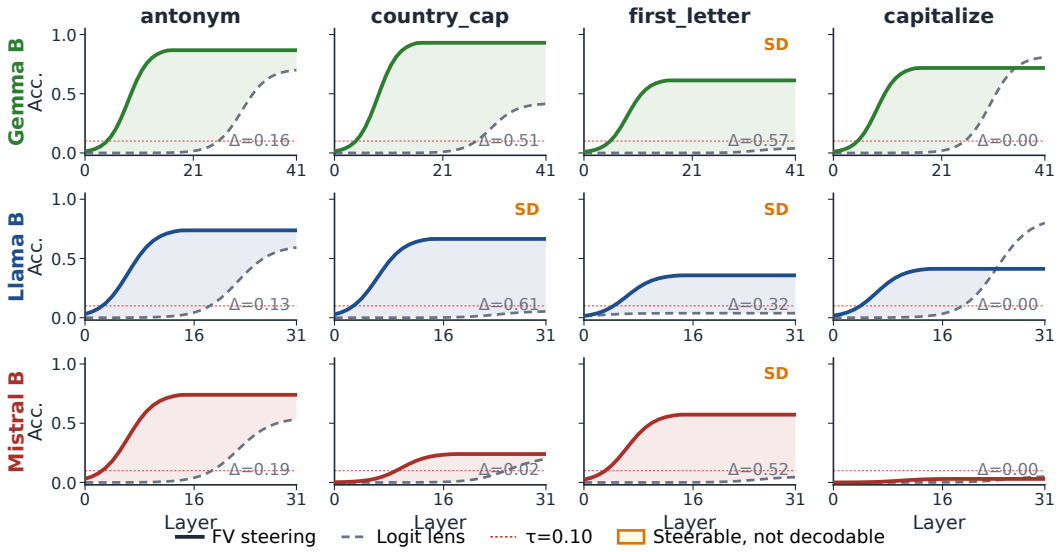

Figure 3: Layer-wise FV steering accuracy (blue) vs. logit-lens top-10 (green dashed) for 4 representative tasks across 3 model families. Pink shading marks the steerability–decodability gap; yellow borders mark steerable-not-decodable panels. FV steering peaks at early layers (L2–L8); logit-lens readability emerges only at late layers (L28–L32). Full 12-panel version: Appendix P.

FVs intervene optimally at early layers (mean L5–L11 across tasks); logit-lens decodability emerges at late layers (L22–L42). For tasks where both work the temporal gap is ∼10 layers; for steerable-not-decodable tasks like `country_capital` on Llama IT the gap is 24+ layers (FV at L5.2, first decodable at L30). The FV injects a perturbation that the model's subsequent 20+ layers convert into the correct output, but no intermediate stage produces a logit-lens-readable answer, consistent with the FV encoding a process rather than an answer.

5.3 Tuned-Lens Validation

To rule out representational dialect [5] we apply per-layer diagonal affine translators (§3). Across all 6 models we identify 14 task×model pairs in the steerable-not-decodable cell under the logit lens; after dialect correction, 13 of 14 (93%) persist (Table 2). The single closure is `first_letter` on Llama IT, a borderline case crossing $\tau = 0.10$ ($0.047 \rightarrow 0.128$); for the remaining 13, the mean tuned-lens delta is -0.005 ($t = -1.71$, $p = 0.113$, n.s.). The most dramatic case is unchanged: `country_capital` on Llama Base reaches 0.880 FV steering while both lenses detect 0.056. Dialect-correction magnitude varies $\sim$10$\times$ across families (Gemma 2.8%, Llama 10.8%, Mistral 24.9%) yet all three converge on the same dissociation. FV vocabulary projection coherence does not improve under dialect correction (mean correct-output fraction $0.018 \rightarrow 0.023$); the dialect anti-correlation analysis ($r = -0.478$ between correction magnitude and readability gain) is in Appendix L.

Table 2: Gap persistence under tuned-lens dialect correction. "S&ND" = steerable-not-decodable pairs under the logit lens. "Closed" = tuned-lens top-10 crosses $\tau = 0.10$.

Model	S&ND tasks	Closed	Persist
Llama-3.1 Base	5	0	5
Llama-3.1 Instruct	1	1	0
Mistral Base	0	0	0
Mistral Instruct	4	0	4
Gemma-2 Base	1	0	1
Gemma-2 Instruct	3	0	3
Total	**14**	**1 (7%)**	**13 (93%)**

5.4 Nonlinear-Decoder Follow-up

A diagonal tuned lens cannot capture nonlinear structure. We train a 2-layer MLP probe (LayerNorm $\rightarrow$ Linear($d \rightarrow 1024$) $\rightarrow$ GELU $\rightarrow$ Dropout $\rightarrow$ Linear($1024 \rightarrow |V|$)) on pooled zero-shot residuals with an 80/20 split by unique input. Every probe is paired with a Hewitt & Liang control [16] trained against a deterministic shuffle of the input$\rightarrow$label mapping; we report selectivity = real top-10 $-$ control top-10 and gate decodability at selectivity $\geq \tau/2 = 0.05$. Full design: Appendix M.

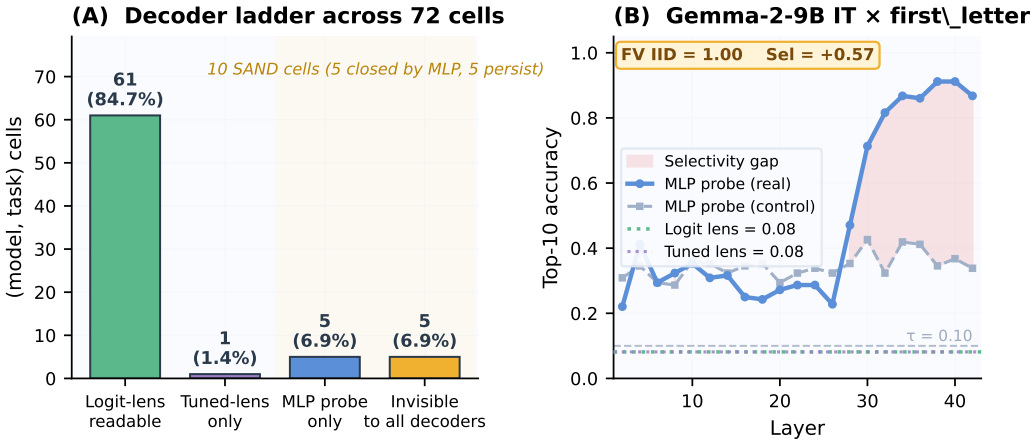

Figure 4: Nonlinear-decoder follow-up. **(A)** Decoder ladder across 72 (model, task) cells: 61 are decodable by the logit lens, 1 by the tuned lens only, 5 by the MLP probe only (closing 5 of 10 SAND cases), and 5 remain invisible to every decoder we tried. **(B)** The cleanest closure: Gemma-2-9B IT $\times$ `first_letter`. Logit lens 0.081 misses every layer; the MLP probe reaches 0.912 top-10 at L38 with selectivity $+0.57$, while FV steering reaches 1.00.

The MLP probe closes 5 of 10 logit-lens SAND cells: Llama Base $\times$ {`country_capital`, `hypernym`}, Llama IT $\times$ `first_letter`, Gemma IT $\times$ `first_letter`, and Mistral IT $\times$

country_capital. For these the FV operates through nonlinearly encoded subspaces orthogonal to W_U. The cleanest closure is Gemma-2-9B IT $\times$ first_letter: logit lens 0.081, MLP probe 0.912 at L38, selectivity $+0.57$, FV steering 1.00 (Figure 4B). The remaining 5 SAND cells survive: Llama Base $\times$ {first_letter, object_color}, Gemma IT $\times$ object_color, Mistral IT $\times$ {object_color, synonym}. For these, FV steering succeeds (IID 0.46–0.90) while every decoder we tried fails to recover input-conditional structure.

The Hewitt & Liang gate vetoes 6 false positives on object_color where probes attain 0.82–0.88 top-10 by exploiting the small-alphabet label marginal (red, blue, green, . . .); without the gate the conclusion "the FV reveals object_color information" would be a textbook probe-shortcut artifact. Symmetrically, the highest selectivities (hypernym $+0.74$ on Gemma Base, first_letter $+0.57$ on Gemma IT) are confidently nonlinear positive findings. Per-task selectivity bars, layer-wise patterns, and the (logit-lens, MLP-real) scatter are in Appendix M.

5.5 Post-Steering Logit Lens: Model-Family Mechanistic Divergence

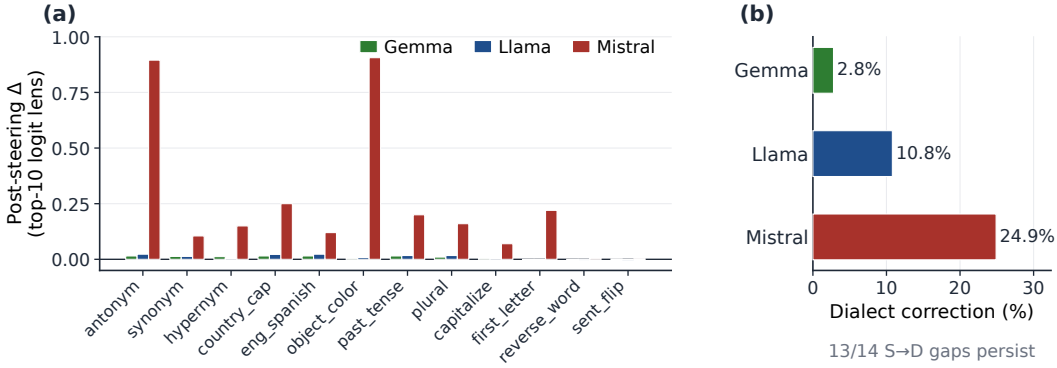

Figure 5: Model-family mechanistic divergence. **(A)** Maximum post-steering logit-lens delta per task: Mistral Instruct FVs massively rewrite intermediate representations (delta up to $+1.000$); Llama and Gemma FVs produce near-zero changes on steerable-not-decodable tasks despite successful steering. **(B)** Dialect correction magnitude by family: despite a $\sim 10\times$ range, 93% of SAND gaps persist under the tuned lens.

Applying the logit lens to post-steering activations exposes a sharp family split (Figure 5). On Llama and Gemma, FV injection on steerable-not-decodable tasks barely changes logit-lens readings (e.g., Llama Base first_letter: max post-steering top-10 is 0.035; country_capital: 0.078): the FV steers the final output without leaving an intermediate trace. On Mistral the same task category shows massive rewriting: Mistral IT object_color jumps from 0.012 (zero-shot) to 0.918 post-steering (delta $+0.906$); antonym reaches $+1.000$. Post-steering delta correlates strongly with IID steering accuracy ($r = 0.52$–0.88 across 6 models), far exceeding cosine alignment's predictive power ($|r| < 0.20$ within-task).

Three independent lines of evidence support a dual-mechanism interpretation, in which Mistral FVs operate representationally (writing into the subspace W_U reads) while Llama and Gemma FVs operate modulatorily (altering attention routing or MLP gating without changing logit-lens-visible representations): (i) post-steering deltas, (ii) activation-patching recovery rates (§5.6), and (iii) FV norm–transfer correlations (only Mistral IT positive; Appendix I). This asymmetry resonates with circuit-level work distinguishing MLPs (writing in vocabulary space [11]) from attention (routing rather than writing [12]). Per-task deltas: Appendix N.

5.6 Activation Patching: Causal Localization

Patching the residual stream from a corrupted (wrong-template FV) to a clean (correct-template FV) run and measuring accuracy recovery [22] reveals a sharp task-dependent pattern (full table: Appendix O). Mean recovery is low (0.10–0.30), but tasks with strong IID steering (past_tense, plural, first_letter, antonym) achieve perfect recovery (1.000) on 4–5 of 6 models at specific

layers; tasks where FV extraction genuinely fails (`reverse_word`, `sentiment_flip`) show zero recovery on 5 of 6 models at every layer. Peak recovery layers overlap substantially with optimal FV-injection layers, supporting the interpretation that FVs redirect computation at specific processing bottlenecks. Llama models show higher mean recovery (0.275–0.295) than Gemma IT (0.101) despite Gemma IT achieving higher IID accuracy, providing the third independent line of evidence for the dual-mechanism hypothesis.

6 Discussion

Decomposing the linear representation hypothesis. Park et al. [27] formalize two faces of the LRH, an unembedding representation tied to probing and an embedding representation tied to additive steering, and unify them via a causal inner product. Our results provide an empirical, model-internal counterpart at the level of the parameter-free logit lens, decomposing the LRH into linear decodability and linear steerability and showing they come apart in the opposite direction from naive intuition. This is consistent with reports that early-to-mid-layer steering vectors do not project to target logits [32, 38], that input and output features rarely co-occur at the SAE level [2], and that steering can operate via dormant pathways [20]. Our contribution is the systematic multi-task, multi-model, tuned-lens- and MLP-probe-validated quantification.

Implications for AI safety. (i) Steering effects may be invisible to standard monitoring: FVs steer behavior through pathways the logit lens cannot detect, especially on Llama and Gemma; safety monitoring relying on vocabulary-projection tools may miss FV-style interventions. (ii) FV vocabulary projection is not a reliable quality check: FVs achieving >0.90 accuracy project to incoherent tokens. (iii) Verify IID efficacy before deployment: IID accuracy is the single best predictor of FV quality; cosine similarity and vocabulary projection are unreliable.

Limitations. Tasks require single-token or short outputs; open-ended generation may differ. All models are 7–9B; larger models may develop more unembedding-aligned representations. We evaluate only mean-difference extraction. The diagonal tuned lens cannot capture off-diagonal rotations; the 2-layer MLP probe leaves room for higher-capacity nonlinear decoders, though the Hewitt & Liang gate guarantees any future closure will not be a label-marginal artifact. Eight templates per task, while substantially more than prior work, do not cover adversarial phrasings. Directly testing the dual-mechanism hypothesis via attention-head patching is a concrete next step.

7 Conclusion

We tested whether the logit lens can explain FV steering failures across 12 tasks, 6 models, and 4,032 directed pairs, and found a steerability-without-decodability inversion: FV steering succeeds even when the model's own unembedding cannot decode the correct output at any layer (gaps as large as -0.91), while the predicted opposite pattern is nearly empty (3/72). Per-layer diagonal tuned-lens correction closes 1 of 14 SAND gaps (93% persist); a 2-layer MLP probe with a Hewitt & Liang control closes 5 of 10 cells while leaving 5 invisible to every decoder we tried. Activation patching causally confirms the difficulty hierarchy; post-steering analysis exposes a model-family mechanistic divergence; the previously reported negative cosine–transfer correlation dissolves at scale. Practitioners should verify IID efficacy before deploying steering vectors and should not rely on logit-lens or vocabulary-projection tools to validate or monitor steering effects: FVs operate through subspaces these tools are blind to.

References

[1] Nadaf, S. Cross-template steering vector transfer reveals task-dependent failures of the linear representation assumption. Blog post, 2025. `https://suhailnadaf509.github.io/blog/simpsons-paradox-fv-extraction`.

[2] Arad, D., Mueller, A., and Belinkov, Y. SAEs are good for steering—if you select the right features. *EMNLP*, 2025.

[3] Arditi, A., Obeso, O., Shlegeris, B., and Amodei, D. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.11717*, 2024.

[4] Belinkov, Y. Probing classifiers: promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

[5] Belrose, N., Furman, Z., Smith, L., Halawi, D., Ostrovsky, I., McKinney, L., Biderman, S., and Steinhardt, J. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

[6] Braun, J., Bloom, J., et al. Understanding (un)reliability of steering vectors in language models. *ICLR Workshop*, 2025.

[7] Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., et al. Towards monosemanticity: decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

[8] Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. *NeurIPS*, 2023.

[9] Dar, G., Geva, M., Gupta, A., and Berant, J. Analyzing transformers in embedding space. *ACL*, 2023.

[10] Da Silva, P., et al. Steering off course: reliability challenges in steering language models. *ACL*, 2025.

[11] Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. *EMNLP*, 2022.

[12] Geva, M., Caciularu, A., Wang, K., and Goldberg, Y. Dissecting recall of factual associations in auto-regressive language models. *EMNLP*, 2023.

[13] Goldman-Wetzler, J. and Turner, A. M. I found >800 orthogonal "write code" steering vectors. *MATS / LessWrong*, 2024.

[14] Hendel, R., Geva, M., and Globerson, A. In-context learning creates task vectors. *EMNLP Findings*, 2023.

[15] Hernandez, E., Li, B. Z., and Andreas, J. Linearity of relation representations in transformer language models. *ICLR*, 2024.

[16] Hewitt, J. and Liang, P. Designing and interpreting probes with control tasks. *EMNLP*, 2019.

[17] Im, S. and Li, Y. A unified understanding and evaluation of steering methods. *arXiv preprint arXiv:2502.02716*, 2025.

[18] Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: eliciting truthful answers from a language model. *NeurIPS*, 2024.

[19] Liu, T., Guo, S., and Arora, S. In-context vectors: making in-context learning more effective and controllable through latent space steering. *arXiv preprint arXiv:2311.06668*, 2023.

[20] Makelov, A., Lange, G., and Nanda, N. Is this the subspace you are looking for? An interpretability illusion for subspace activation patching. *ICLR*, 2024.

[21] Marks, S. and Tegmark, M. The geometry of truth: emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2023.

[22] Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. *NeurIPS*, 2022.

[23] nostalgebraist. Interpreting GPT: the logit lens. *LessWrong*, 2020. `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`.

[24] Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., et al. In-context learning and induction heads. *Transformer Circuits Thread*, 2022.

[25] Panickssery, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering Llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2023.

[26] Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

[27] Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. *ICML*, 2024.

[28] Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering Llama 2 via contrastive activation addition. *ACL*, 2024.

[29] Sclar, M., Choi, Y., Tsvetkov, Y., and Suhr, A. Quantifying language models' sensitivity to spurious features in prompt design. *arXiv preprint arXiv:2310.11324*, 2023.

[30] She, J., et al. Linear steerability in language models: decodability before steerability. *EMNLP Findings*, 2025.

[31] Simpson, E. H. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society: Series B*, 13(2):238–241, 1951.

[32] Sinii, V., et al. Steering LLM reasoning through bias-only adaptation. *EMNLP*, 2025.

[33] Tan, D., Chanin, D., Lynch, A., Kanoulas, D., Paige, B., Garriga-Alonso, A., and Kirk, R. Analysing the generalisation and reliability of steering vectors. *NeurIPS*, 2024.

[34] Tigges, C., Hollinsworth, O. J., Geiger, A., and Nanda, N. Linear representations of sentiment in large language models. *arXiv preprint arXiv:2310.15154*, 2023.

[35] Todd, E., Li, M. L., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. Function vectors in large language models. *ICLR*, 2024.

[36] Turner, A. M., Thiergart, L., Udell, D., Leech, G., Mini, U., and MacDiarmid, M. Activation addition: steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.

[37] Wang, K., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. *ICLR*, 2023.

[38] Ward, F., Lin, J., Venhoff, C., and Nanda, N. Reasoning-finetuning repurposes latent representations in base models. *MATS / LessWrong*, 2025.

[39] Webson, A. and Pavlick, E. Do prompt-based models really understand the meaning of their prompts? *NAACL*, 2022.

[40] Wei, A., Haghtalab, N., and Steinhardt, J. Jailbroken: how does LLM safety training fail? *NeurIPS*, 2024.

[41] Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., et al. Representation engineering: a top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

Appendix Contents

The appendix consists of three parts. **Methodology details:** task taxonomy and template specifications (App. A), extended IID and baseline tables (App. B), per-template breakdowns (App. C), within-style vs. across-style transfer (App. D), instruction-tuning effects (App. J), full transfer tables (App. E). **Geometric and dissociation analyses:** per-model within-task correlations (App. F), permutation tests (App. G), universal-template-vector PCA (App. H), FV norms (App. I), pre-registered predictions (App. K). **Inversion mechanics:** tuned-lens statistics and dissociation matrix (App. L), MLP-probe decoder ladder (App. M), post-steering logit-lens details (App. N), activation patching results (App. O), additional figures (App. P), statistical and computational details (App. Q, App. R), and a worked-example walkthrough (App. S). The appendix is free-standing: a reader who has not seen our codebase can re-implement the methodology from these specifications alone.

A Complete Template Specifications

Table 3 summarizes the 12-task battery; Table 4 lists all 8 templates for all 12 tasks (96 templates total). Templates T1–T2 are Natural style, T3–T4 Symbolic, T5–T6 Question, and T7–T8 Formal. Every template contains at least one task-specific keyword, and symbolic templates use unique function names; no two tasks share any template string.

Task taxonomy and motivation. The 5 categories correspond to qualitatively different computational demands: **Lexical Retrieval** (stored word-level associations), **Factual Retrieval** (world-knowledge facts), **Morphological Transform** (regular grammatical rules), **Character/Surface** (character-level manipulation; predicted to fail), and **Compositional/Semantic** (multi-token semantic rewriting; predicted to fail). The taxonomy is motivated by the hypothesis that different computation types engage different transformer circuits [12, 35] and may exhibit different FV geometry and transfer behavior.

Table 3: Task battery: 12 tasks across 5 computational categories. "Expected IID" is the pre-registered difficulty prediction (App. K), formulated before running any experiments. n is the number of input–output pairs. Eval: substring match (case-sensitive for `capitalize`); polarity classification for `sentiment_flip` (App. Q). Tok: maximum new tokens.

Category	Task	n	Eval	Expected IID	Role	Tok
Lexical Retrieval	`antonym`	95	substr	0.45–0.60	Positive control	5
	`synonym`	88	substr	0.20–0.40	Moderate test	5
	`hypernym`	86	substr	0.25–0.45	Moderate test	5
Factual Retrieval	`country_capital`	90	substr	0.40–0.65	Positive control	5
	`english_spanish`	88	substr	0.25–0.50	Moderate test	5
	`object_color`	85	substr	0.30–0.55	Moderate test	5
Morphological	`past_tense`	90	substr	0.35–0.55	Positive control	5
	`plural`	90	substr	0.30–0.50	Moderate test	5
Character / Surface	`capitalize`	84	case-s.	0.00–0.05	Negative control	5
	`first_letter`	86	substr	0.05–0.20	Boundary test	3
	`reverse_word`	80	substr	0.00–0.08	Negative control	5
Compositional	`sentiment_flip`	60	substr	0.00–0.05	Negative control	10

Definition 1 (IID gating)**.** *A task t on model m* passes the IID gate *if its mean IID accuracy across templates exceeds $\tau = 0.10$. Tasks failing the gate are excluded from causal analyses (activation patching), since for these tasks FV extraction itself has not produced a working steering direction.*

Table 4: Complete template specifications for all 12 tasks. Curly braces enclose the input slot.

Task	ID	Style	Template
`antonym`	T1	Natural	The opposite of {X} is
`antonym`	T2	Natural	{X} has the opposite meaning of
`antonym`	T3	Symbolic	antonym({X}) =
`antonym`	T4	Symbolic	opposite: {X} →
`antonym`	T5	Question	What is the antonym of {X}?
`antonym`	T6	Question	What word means the opposite of {X}?
`antonym`	T7	Formal	Antonym relation: {X} maps to
`antonym`	T8	Formal	Given the word {X}, the antonym is
`synonym`	T1	Natural	A synonym for {X} is
`synonym`	T2	Natural	Another word for {X} is
`synonym`	T3	Symbolic	synonym({X}) =
`synonym`	T4	Symbolic	similar_word: {X} →
`synonym`	T5	Question	What is a synonym of {X}?
`synonym`	T6	Question	What word has a similar meaning to {X}?
`synonym`	T7	Formal	Synonym identification: {X} corresponds to
`synonym`	T8	Formal	The word {X} is synonymous with
`hypernym`	T1	Natural	{X} is a type of
`hypernym`	T2	Natural	The category of {X} is
`hypernym`	T3	Symbolic	hypernym({X}) =
`hypernym`	T4	Symbolic	category: {X} →
`hypernym`	T5	Question	What kind of thing is a {X}?
`hypernym`	T6	Question	What category does {X} belong to?

Continued on next page

Task	ID	Style	Template
hypernym	T7	Formal	Taxonomic classification: {X} is classified as
hypernym	T8	Formal	In terms of hierarchy, {X} falls under
country_capital	T1	Natural	The capital of {X} is
country_capital	T2	Natural	{X}'s capital city is
country_capital	T3	Symbolic	capital({X}) =
country_capital	T4	Symbolic	country_capital: {X} →
country_capital	T5	Question	What is the capital of {X}?
country_capital	T6	Question	Which city is the capital of {X}?
country_capital	T7	Formal	Capital city identification: {X} has capital
country_capital	T8	Formal	For the country {X}, the capital is
english_spanish	T1	Natural	The Spanish word for {X} is
english_spanish	T2	Natural	In Spanish, {X} is called
english_spanish	T3	Symbolic	translate_es({X}) =
english_spanish	T4	Symbolic	en_to_es: {X} →
english_spanish	T5	Question	How do you say {X} in Spanish?
english_spanish	T6	Question	What is the Spanish translation of {X}?
english_spanish	T7	Formal	English-Spanish translation: {X} renders as
english_spanish	T8	Formal	The Spanish equivalent of {X} is
object_color	T1	Natural	The color of a {X} is
object_color	T2	Natural	A {X} is typically colored
object_color	T3	Symbolic	color({X}) =
object_color	T4	Symbolic	object_color: {X} →
object_color	T5	Question	What color is a {X}?
object_color	T6	Question	What is the typical color of a {X}?
object_color	T7	Formal	Color association: {X} corresponds to
object_color	T8	Formal	The characteristic color of {X} is
past_tense	T1	Natural	The past tense of {X} is
past_tense	T2	Natural	Yesterday I {X}, so I
past_tense	T3	Symbolic	past_tense({X}) =
past_tense	T4	Symbolic	verb_past: {X} →
past_tense	T5	Question	What is the past tense of {X}?
past_tense	T6	Question	How do you conjugate {X} in the past?
past_tense	T7	Formal	Past tense conjugation: {X} becomes
past_tense	T8	Formal	The simple past form of {X} is
plural	T1	Natural	The plural of {X} is
plural	T2	Natural	{X} in plural form is
plural	T3	Symbolic	plural({X}) =
plural	T4	Symbolic	noun_plural: {X} →
plural	T5	Question	What is the plural of {X}?
plural	T6	Question	How do you pluralize {X}?
plural	T7	Formal	Plural formation: {X} becomes
plural	T8	Formal	The plural form of the noun {X} is
capitalize	T1	Natural	{X} in uppercase is
capitalize	T2	Natural	The uppercase version of {X} is
capitalize	T3	Symbolic	UPPERCASE({X}) =
capitalize	T4	Symbolic	to_upper: {X} →
capitalize	T5	Question	What is {X} in all capital letters?
capitalize	T6	Question	How do you write {X} in uppercase?
capitalize	T7	Formal	Uppercase conversion: {X} becomes
capitalize	T8	Formal	Applying capitalization to {X} yields
first_letter	T1	Natural	The first letter of {X} is
first_letter	T2	Natural	{X} starts with the letter
first_letter	T3	Symbolic	first_char({X}) =
first_letter	T4	Symbolic	initial: {X} →
first_letter	T5	Question	What letter does {X} start with?
first_letter	T6	Question	What is the first letter of {X}?
first_letter	T7	Formal	Initial letter extraction: {X} yields
first_letter	T8	Formal	The leading character of {X} is

Continued on next page

Task	ID	Style	Template
`reverse_word`	T1	Natural	{X} spelled backwards is
`reverse_word`	T2	Natural	The reverse of {X} is
`reverse_word`	T3	Symbolic	reverse({X}) =
`reverse_word`	T4	Symbolic	reversed: {X} →
`reverse_word`	T5	Question	What is {X} spelled in reverse?
`reverse_word`	T6	Question	How do you spell {X} backwards?
`reverse_word`	T7	Formal	String reversal: {X} becomes
`reverse_word`	T8	Formal	Reversing the characters of {X} yields
`sentiment_flip`	T1	Natural	Rewrite with opposite sentiment: {X} becomes
`sentiment_flip`	T2	Natural	The negative version of {X} is
`sentiment_flip`	T3	Symbolic	flip_sentiment({X}) =
`sentiment_flip`	T4	Symbolic	sentiment_reverse: {X} →
`sentiment_flip`	T5	Question	What is the opposite sentiment of {X}?
`sentiment_flip`	T6	Question	How would you express {X} negatively?
`sentiment_flip`	T7	Formal	Sentiment inversion: {X} transforms to
`sentiment_flip`	T8	Formal	Applying sentiment reversal to {X} yields

Dataset construction. Each task's input–output pairs are curated lists of single-token-output items ($n \in [60, 95]$; counts in Table 3). Pairs are validated to ensure (i) at least 60 pairs (so that the experiment can use 15 ICL positives and 50 IID test items), (ii) no duplicate inputs, and (iii) every template contains the input placeholder. Expected outputs are chosen so that their first sub-word token identifies the answer under all three tokenizer families (Llama BPE, Gemma SentencePiece, Mistral BPE). For each candidate expected output, we encode it both as-is and with a leading space prefix and use whichever produces a valid first token; this is necessary because the three tokenizer families treat the leading-space variant differently (e.g., Llama and Mistral BPE typically use a leading-space token for words that follow a space, while Gemma SentencePiece may use the bare token). All 12 tasks pass this validation under all three tokenizers.

For each (task, template) pair, we produce four sets of prompts (deterministic under a fixed random seed): (1) **ICL positive prompts** for FV extraction: 15 inputs, each formatted as 5 few-shot demonstrations followed by the test input using the same template string; demonstrations are sampled from a seeded-shuffled pool that excludes the test input; (2) **ICL negative prompts**: the same 15 inputs as bare zero-shot prompts; (3) **IID test prompts**: 50 bare prompts using the same template as extraction; (4) **OOD test prompts**: 50 bare prompts per other template, yielding 7 OOD sets per source template. Each model thus processes approximately $12 \times 8 \times 430 \approx 41{,}280$ prompts; activations are cached once per (task, template) to amortize the α and layer sweep.

Mean-of-differences as a contrastive estimator. The mean-of-differences FV $\mathrm{FV} = \mathbb{E}[\mathbf{h}^{(\mathrm{pos})}] - \mathbb{E}[\mathbf{h}^{(\mathrm{neg})}]$ in Eq. (1) is the standard CAA estimator [17, 25, 28]. Under the assumption that positive and negative classes have approximately equal-covariance Gaussian activations, this is the maximum-likelihood linear-discriminant direction up to scale; under the linear representation hypothesis, it is the projection onto the task-relevant direction. Im & Li [17] compared CAA, PCA, and classifier-based extraction and found mean-of-differences theoretically and empirically strong; we adopt it as the standard formulation.

B Extended IID Accuracy and Baseline Tables

Tables 5, 6, and 7 report zero-shot, 5-shot ICL, and IID-steering accuracy per task per family. Zero-shot baseline accuracy correlates strongly with IID steering across all 6 models: Gemma Base $r = 0.79$, Gemma IT $r = 0.75$, Llama Base $r = 0.87$, Llama IT $r = 0.73$, Mistral Base $r = 0.87$, Mistral IT $r = 0.81$ (all $p < 10^{-3}$). FV steering redirects existing capabilities; it does not teach new ones.

Aggregate destructive cases. Across all 6 models, the rate at which the mean steered accuracy drops below the zero-shot baseline (FV steering hurts the task in aggregate) is 0–1%. As discussed in §4, this aggregate-task statistic is compatible with prior reports of substantial per-sample anti-steerability [6, 33]: positive effects can dominate the mean even when individual examples are pushed

Table 5: Baselines and IID accuracy for Llama-3.1-8B Base and Instruct. Zero-shot and 5-shot ICL are measured independently of FV extraction; IID is the mean across 8 templates at the best layer and best α. Bold-red entries fail the IID gate ($\tau = 0.10$).

		Llama Base			Llama Instruct		
Cat.	**Task**	Zero	5-shot	IID	Zero	5-shot	IID
Lex.	antonym	0.46	0.76	0.738	0.56	0.86	0.870
	synonym	0.20	0.58	0.357	0.22	0.51	0.625
	hypernym	0.04	0.09	**0.085**	0.08	0.37	0.412
Fact.	country_capital	0.08	0.84	0.665	0.14	0.86	0.920
	english_spanish	0.56	0.86	0.833	0.64	0.85	0.882
	object_color	0.03	0.49	0.357	0.09	0.50	0.720
Morph.	past_tense	0.50	0.96	0.892	0.43	0.96	0.943
	plural	0.45	0.88	0.845	0.51	0.89	0.983
Char.	capitalize	0.07	0.86	0.413	0.23	0.94	0.725
	first_letter	0.17	0.26	0.357	0.27	0.38	0.777
	reverse_word	0.00	0.07	**0.028**	0.05	0.47	0.185
Comp.	sentiment_flip	0.04	0.05	**0.068**	0.06	0.03	0.140

Table 6: Baselines and IID accuracy for Gemma-2-9B Base and Instruct. Format matches Table 5.

		Gemma-2 Base			Gemma-2 IT		
Cat.	**Task**	Zero	5-shot	IID	Zero	5-shot	IID
Lex.	antonym	0.59	0.68	0.867	0.63	0.68	0.880
	synonym	0.17	0.48	0.440	0.29	0.57	0.675
	hypernym	0.21	0.92	0.902	0.14	0.88	0.897
Fact.	country_capital	0.42	1.00	0.930	0.53	1.00	0.982
	english_spanish	0.61	1.00	0.897	0.62	0.99	0.905
	object_color	0.09	0.12	0.522	0.06	0.19	0.575
Morph.	past_tense	0.56	1.00	0.948	0.57	1.00	0.982
	plural	0.65	1.00	0.998	0.71	1.00	1.000
Char.	capitalize	0.06	1.00	0.718	0.40	1.00	0.865
	first_letter	0.45	0.36	0.613	0.41	0.15	0.748
	reverse_word	0.04	0.28	**0.085**	0.34	0.26	0.393
Comp.	sentiment_flip	0.02	0.05	0.125	0.03	0.01	0.198

Table 7: Baselines and IID accuracy for Mistral-7B-v0.3 Base and Instruct. Mistral zero-shot accuracies are notably lower than Llama and Gemma for lexical tasks, reflecting weaker few-shot formatting sensitivity.

		Mistral Base			Mistral Instruct		
Cat.	**Task**	Zero	5-shot	IID	Zero	5-shot	IID
Lex.	antonym	0.02	0.68	0.740	0.02	0.66	0.780
	synonym	0.01	0.16	0.138	0.09	0.18	0.367
	hypernym	0.09	0.12	0.315	0.07	0.12	0.435
Fact.	country_capital	0.14	0.18	0.240	0.02	0.10	0.265
	english_spanish	0.59	0.76	0.818	0.60	0.77	0.797
	object_color	0.12	0.28	0.578	0.02	0.28	0.688
Morph.	past_tense	0.51	0.93	0.823	0.67	0.93	0.895
	plural	0.58	0.85	0.940	0.83	0.85	0.983
Char.	capitalize	0.01	0.08	**0.030**	0.03	0.08	**0.088**
	first_letter	0.23	0.43	0.573	0.11	0.30	0.723
	reverse_word	0.00	0.15	**0.030**	0.01	0.07	**0.037**
Comp.	sentiment_flip	0.03	0.03	**0.045**	0.00	0.00	0.095

in the wrong direction. The aggregate result establishes that FV injection at the task-mean level is consistently additive or neutral; it does not contradict the existence of per-sample failures documented in prior work.

C Per-Template IID Accuracy

Per-template variance is substantial within several tasks. Table 8 shows per-template IID accuracy for Llama-3.1-8B Base. `antonym` ranges from 0.10 (T2) to 0.92 (T8); `country_capital` from 0.22 (T2) to 0.88 (T4); `plural` achieves 1.00 on 6 of 8 templates. Per-template variance is one source of IID–OOD gap heterogeneity: a strong source template transfers well, a weak one transfers poorly.

Table 8: Per-template IID accuracy for Llama-3.1-8B Base. Each cell is the best accuracy across all α values and layers for the (task, template) pair.

Task	T1	T2	T3	T4	T5	T6	T7	T8	**Mean**
`antonym`	.90	.10	.60	.88	.76	.86	.88	.92	.738
`synonym`	.54	.60	.04	.28	.38	.42	.32	.28	.357
`hypernym`	.14	.02	.40	.00	.02	.02	.04	.04	.085
`country_capital`	.86	.22	.34	.88	.58	.76	.84	.84	.665
`english_spanish`	.94	.92	.56	.90	.72	.80	.90	.92	.833
`object_color`	.34	.48	.28	.32	.30	.40	.38	.36	.357
`past_tense`	1.00	.68	.62	1.00	.90	.96	.98	1.00	.892
`plural`	1.00	.16	.60	1.00	1.00	1.00	1.00	1.00	.845
`capitalize`	.00	.90	.16	.48	.46	.40	.66	.24	.413
`first_letter`	.44	.32	.46	.34	.30	.28	.36	.36	.357
`reverse_word`	.12	.00	.00	.00	.06	.00	.02	.02	.028
`sentiment_flip`	.14	.06	.06	.08	.04	.04	.08	.04	.068

D Within-Style vs. Across-Style Transfer

Table 9 reports within-style and across-style mean OOD transfer accuracies for all 6 models. The tested null hypothesis is that within-style pairs (e.g., Natural→Natural) achieve higher transfer than across-style pairs (e.g., Natural→Symbolic). It is not rejected on any model: $p > 0.30$ in every case. This is a positive finding—a quantitative confirmation, with formal statistical tests, of the qualitative observation by Todd et al. [35] that FVs are "remarkably robust" across surface contexts and the finding of Braun et al. [6] that no prompt type consistently dominates: FVs capture an abstract, style-invariant computational instruction not tied to the surface form of the extraction template.

Table 9: Within-style vs. across-style mean OOD transfer accuracy. All 6 models show no significant style effect ($p > 0.30$).

Model	**Within-style mean**	**Across-style mean**	t	p
Gemma-2 Base	0.613	0.644	-0.84	0.40
Gemma-2 IT	0.723	0.735	-0.35	0.72
Llama-3.1 Base	0.424	0.438	-0.36	0.72
Llama-3.1 IT	0.663	0.663	$+0.01$	0.99
Mistral Base	0.400	0.416	-0.42	0.67
Mistral IT	0.456	0.478	-0.56	0.58

E Cross-Template Transfer per Model

Table 10 reports the source-IID-vs-OOD-transfer correlation across all 6 models, plus the IID–OOD gap aggregated over IID-gated tasks. The strong source-IID–transfer correlation ($r = 0.80$–0.93) confirms that cross-template "failure" is primarily a source-quality problem: weak source templates produce weak FVs, which transfer weakly to all targets; strong source templates transfer well across all 7 target templates.

Table 10: Source-template IID quality strongly predicts OOD transfer accuracy across all 6 models. "Mean OOD gap" is averaged over tasks passing the IID gate ($\tau = 0.10$). Negative gap indicates OOD is below IID; positive gap indicates the opposite (e.g., `plural` on Gemma where OOD slightly exceeds IID due to template-specific noise).

Model	r(**source IID**, **OOD**)	**Mean IID–OOD gap**	**Pairs analyzed**
Llama-3.1 Base	$+0.87$	-0.043	4,616
Llama-3.1 Instruct	$+0.81$	-0.039	5,352
Gemma-2 Base	$+0.93$	-0.024	6,099
Gemma-2 Instruct	$+0.91$	-0.019	5,352
Mistral Base	$+0.80$	-0.058	4,044
Mistral Instruct	$+0.85$	-0.041	4,417

F Geometric Analysis Details

Tables 11, 12, and 13 report per-task alignment–transfer correlations for all 6 models. The pattern in the main text—pooled r near zero (or weakly negative on Mistral), with within-task r scattered around zero—is consistent across families. Mistral IT is the most informative case: pooled $r = -0.187$, yet mean within-task $r = +0.139$, with `object_color` driving a strong positive within-task correlation ($r = +0.589$). This is a textbook Simpson's paradox.

Table 11: Within-task alignment–transfer correlations (Llama-3.1-8B Base). Pooled $r = +0.013$ ($p = 0.16$, not significant). The hypernym within-task correlation ($r = -0.510$) is a floor-effect artifact: hypernym fails the IID gate on Llama Base (mean IID 0.085), so its within-task variation is dominated by noise.

Cat.	**Task**	r	p
Lex.	`antonym`	$+0.053$	0.116
Lex.	`synonym`	-0.008	0.800
Lex.	`hypernym`	-0.510	2.3×10^{-60}
Fact.	`country_capital`	$+0.050$	0.137
Fact.	`english_spanish`	-0.118	3.9×10^{-4}
Fact.	`object_color`	$+0.276$	4.2×10^{-17}
Morph.	`past_tense`	$+0.139$	3.0×10^{-5}
Morph.	`plural`	$+0.248$	4.9×10^{-14}
Char.	`capitalize`	$+0.284$	4.3×10^{-18}
Char.	`first_letter`	-0.348	6.8×10^{-27}
Char.	`reverse_word`	-0.030	0.373
Comp.	`sentiment_flip`	$+0.044$	0.185
Pooled		$+0.013$	0.163
Mean within-task r		$+0.007$	—

Geometric dissociation cases. Defining a dissociation as a cross-template pair with cosine alignment > 0.80 but transfer accuracy < 0.40 (a fixed-threshold scheme), Table 14 reports per-model dissociation counts. We additionally verified that the qualitative conclusion—steering accuracy exceeds geometric similarity as a predictor of transfer—is invariant to the threshold scheme. Re-running the analysis with data-derived percentile thresholds (top-20% cosine vs. bottom-50% accuracy) yields the same population-level conclusions on all 6 models, though absolute counts differ.

Table 12: Within-task alignment–transfer correlations for Gemma-2-9B Base and Instruct.

Cat.	Task	Gemma Base r	Gemma Base p	Gemma IT r	Gemma IT p
Lex.	antonym	$+0.142$	1.1×10^{-6}	$+0.340$	3.0×10^{-33}
Lex.	synonym	$+0.007$	0.82	-0.114	8.3×10^{-5}
Lex.	hypernym	$+0.040$	0.17	$+0.009$	0.75
Fact.	country_capital	-0.267	1.2×10^{-20}	-0.036	0.22
Fact.	english_spanish	-0.081	5.4×10^{-3}	$+0.002$	0.95
Fact.	object_color	-0.137	2.4×10^{-6}	$+0.263$	4.0×10^{-20}
Morph.	past_tense	$+0.117$	5.7×10^{-5}	$+0.062$	0.034
Morph.	plural	-0.146	4.8×10^{-7}	-0.035	0.24
Char.	capitalize	$+0.138$	2.1×10^{-6}	$+0.278$	2.4×10^{-22}
Char.	first_letter	-0.028	0.33	$+0.115$	7.6×10^{-5}
Char.	reverse_word	-0.072	0.014	$+0.138$	2.0×10^{-6}
Comp.	sentiment_flip	$+0.075$	0.010	-0.029	0.32
Pooled		$+0.058$	3.9×10^{-12}	$+0.126$	3.0×10^{-51}
Mean within-task r		-0.018	—	$+0.083$	—

Table 13: Within-task alignment–transfer correlations for Mistral-7B-v0.3. Pooled r is negative for both, yet Mistral IT mean within-task $r = +0.139$—a genuine Simpson's paradox.

Cat.	Task	Mistral Base r	Mistral Base p	Mistral IT r	Mistral IT p
Lex.	antonym	$+0.255$	9.3×10^{-15}	$+0.281$	9.6×10^{-18}
Lex.	synonym	$+0.067$	0.045	-0.023	0.50
Lex.	hypernym	-0.461	2.0×10^{-48}	$+0.071$	0.034
Fact.	country_capital	$+0.017$	0.62	$+0.280$	1.3×10^{-17}
Fact.	english_spanish	-0.028	0.41	$+0.035$	0.29
Fact.	object_color	$+0.282$	8.1×10^{-18}	$+0.589$	7.5×10^{-85}
Morph.	past_tense	$+0.080$	0.016	$+0.181$	4.6×10^{-8}
Morph.	plural	$+0.155$	3.0×10^{-6}	$+0.067$	0.044
Char.	capitalize	$+0.063$	0.060	$+0.158$	1.9×10^{-6}
Char.	first_letter	-0.290	7.2×10^{-19}	-0.124	2.1×10^{-4}
Char.	reverse_word	-0.195	3.7×10^{-9}	-0.148	8.7×10^{-6}
Comp.	sentiment_flip	-0.080	0.017	$+0.296$	1.6×10^{-19}
Pooled		-0.199	1.2×10^{-96}	-0.187	2.0×10^{-85}
Mean within-task r		-0.011	—	$+0.139$	—

Table 14: Geometric dissociation: cross-template pairs with cosine > 0.80 but transfer accuracy < 0.40. "IID-Viable" = dissociations where the source template passes the IID gate, isolating geometry–behavior dissociation from extraction failure.

Model	High-Align Pairs	$N_{\text{dissociation}}$	Rate	IID-Viable
Llama Base	4,470	6,657	0.745	4,616
Llama IT	3,197	2,705	0.423	2,345
Gemma Base	6,351	6,669	0.525	6,099
Gemma IT	5,535	3,860	0.349	3,699
Mistral Base	4,600	7,139	0.776	4,044
Mistral IT	4,212	6,140	0.729	4,417

G Permutation Tests for Excess Dissociation

For each task on each model, we test whether the observed dissociation rate exceeds chance using 1,000 permutations of the alignment–transfer pairing. Per-task p-values are Bonferroni-corrected for the family of 12 within-task tests ($\alpha_{\text{corrected}} = 0.0042$). Significant excess dissociation is task- and model-specific: Gemma Base shows significant excess for 6 of 12 tasks (`country_capital`, `english_spanish`, `hypernym`, `past_tense`, `plural`, `synonym`); Gemma IT for 3 of 12 (`english_spanish`, `reverse_word`, `synonym`); Llama Base for 1 of 12 (`country_capital`); Mistral Base for 1 of 12 (`first_letter`); Llama IT and Mistral IT show no tasks significant after correction. The pattern reflects task-specific geometric structure rather than a uniform artifact.

H Universal Template Vector (UTV) Analysis

We apply PCA to the 8 template FVs per task at the best steering layer and report the fraction of variance explained by PC1. High PC1 indicates templates converge on a shared FV direction; low PC1 indicates templates extract different directions.

Table 15: PC1 variance explained (range across 6 models) for each task. Tasks sorted by mean PC1.

Cat.	**Task**	**PC1 Range**	**Mean PC1**
Char.	`reverse_word`	0.55–0.97	0.80
Lex.	`antonym`	0.52–0.92	0.72
Morph.	`plural`	0.44–0.80	0.63
Char.	`capitalize`	0.42–0.84	0.60
Morph.	`past_tense`	0.38–0.73	0.55
Fact.	`english_spanish`	0.35–0.68	0.50
Fact.	`country_capital`	0.33–0.65	0.48
Lex.	`synonym`	0.37–0.63	0.47
Comp.	`sentiment_flip`	0.30–0.62	0.44
Lex.	`hypernym`	0.32–0.62	0.43
Char.	`first_letter`	0.30–0.58	0.42
Fact.	`object_color`	0.36–0.55	0.42

High PC1 does not predict high transfer: `reverse_word` has the highest template convergence (PC1 = 0.80) yet near-zero IID accuracy on most models (Table 5–7), while `synonym` and `object_color` have low PC1 (0.42–0.47) yet moderate-to-good steering. This confirms that the geometry of template FVs (even their shared direction structure) does not determine functional effectiveness, reinforcing the §4 hierarchical-regression conclusion at a different level of analysis.

I FV Norm Analysis

Table 16 reports mean FV L2 norms and FV-norm–transfer correlations across all cross-template pairs. Gemma FV norms (277–283) are 15–30$\times$ larger than Mistral's (9.7–12.6) and 12–17$\times$ larger than Llama's (16.4–20.5), likely reflecting architectural and normalization differences (Gemma-2's $d_{\text{model}} = 3584$ with 42 layers vs. Llama and Mistral's $d_{\text{model}} = 4096$ with 32 layers, plus differing residual-stream normalization conventions).

Table 16: FV L2 norms (mean across all extracted FVs at the best steering layer) and norm–transfer Pearson correlations across all cross-template pairs ($n \approx$ 10,752–14,112 pairs per model).

Model	**Mean Norm**	r(norm, transfer)	p
Llama-3.1 Base	20.50	-0.089	1.9×10^{-20}
Llama-3.1 IT	16.39	-0.077	1.9×10^{-15}
Gemma-2 Base	283.12	-0.034	6.5×10^{-5}
Gemma-2 IT	277.09	-0.046	3.6×10^{-8}
Mistral Base	9.71	-0.013	0.18
Mistral IT	12.64	$+0.132$	5.8×10^{-43}

The norm–transfer correlation is weakly negative or near zero for 5 of 6 models; only Mistral IT shows a positive relationship ($r = +0.132$). This is the third independent line of evidence (after post-steering logit-lens deltas and patching recovery) for the dual-mechanism hypothesis: Mistral's effect on transfer is genuinely magnitude-driven, while Llama's and Gemma's are not. Notably, despite Gemma's norms being 15–30$\times$ larger than Mistral's, Gemma's per-layer post-steering deltas are smaller (§N). Magnitude is not what drives readability change; alignment with W_U is. This partially refutes a "more push = more change" hypothesis and reinforces the modulatory interpretation of Llama and Gemma FV mechanisms.

J Instruction-Tuning Effect

Table 17 reports the per-task instruction-tuning effect ($\Delta = \text{IT} - \text{Base}$). Instruction tuning consistently improves FV steering accuracy: mean Δ is $+0.21$ (Llama), $+0.09$ (Gemma), $+0.07$ (Mistral). The dissociation pattern is preserved across the IT–Base axis, indicating instruction tuning improves IID efficacy without changing the qualitative conclusion that FV steering operates outside the logit-lens-decodable subspace.

Table 17: Instruction-tuning effect on IID steering accuracy ($\Delta = \text{Instruct} - \text{Base}$).

Task	**Gemma Δ**	**Llama Δ**	**Mistral Δ**
`antonym`	$+0.013$	$+0.132$	$+0.040$
`synonym`	$+0.235$	$+0.268$	$+0.229$
`hypernym`	-0.005	$+0.327$	$+0.120$
`country_capital`	$+0.052$	$+0.255$	$+0.025$
`english_spanish`	$+0.008$	$+0.049$	-0.021
`object_color`	$+0.053$	$+0.363$	$+0.110$
`past_tense`	$+0.034$	$+0.051$	$+0.072$
`plural`	$+0.002$	$+0.138$	$+0.043$
`capitalize`	$+0.147$	$+0.312$	$+0.058$
`first_letter`	$+0.135$	$+0.420$	$+0.150$
`reverse_word`	$+0.308$	$+0.157$	$+0.007$
`sentiment_flip`	$+0.073$	$+0.072$	$+0.050$
Mean	$+0.088$	$+0.212$	$+0.074$

K Pre-Registered Predictions and Outcomes

Table 18: Pre-registered predictions vs. observed grand mean across 6 models. ✓ = confirmed within range; $\sim$ = partially confirmed (slightly above); $\times$ = falsified. Predictions were committed before any model runs.

Task	**Predicted**	**Observed**	**Status**	**Notes**
`antonym`	0.45–0.60	0.813	$\times$	Exceeded range
`synonym`	0.20–0.40	0.434	$\sim$	Slightly above
`hypernym`	0.25–0.45	0.508	$\sim$	Strongly model-dependent
`country_capital`	0.40–0.65	0.667	$\sim$	Slightly above
`english_spanish`	0.25–0.50	0.856	$\times$	Greatly exceeded
`object_color`	0.30–0.55	0.573	$\sim$	Slightly above
`past_tense`	0.35–0.55	0.914	$\times$	Greatly exceeded
`plural`	0.30–0.50	0.958	$\times$	Greatly exceeded
`capitalize`	0.00–0.05	0.473	$\times$	Falsified neg. control
`first_letter`	0.05–0.20	0.632	$\times$	Falsified neg. control
`reverse_word`	0.00–0.08	0.126	$\sim$	Near predicted
`sentiment_flip`	0.00–0.05	0.112	$\sim$	Near predicted

The prediction grid was conservative: 4 of 12 tasks were greatly exceeded (`antonym`, `english_spanish`, `past_tense`, `plural`), and 2 of 3 character "negative controls" (`capitalize`, `first_letter`) were strongly falsified. Only `reverse_word` and `sentiment_flip` behaved as

predicted. The honest assessment of these failures motivated the IID-gating definition (§1): rather than retroactively redefine task difficulty, we use a single threshold to admit tasks into causal analysis.

L Tuned-Lens Statistics

Training procedure. Per-layer translators are trained independently for each layer using AdamW (learning rate 10^{-3}, weight decay 0.01), early stopping with patience 15 epochs and improvement threshold 10^{-7}, and a maximum of 200 epochs. Training data are pooled zero-shot residual-stream activations across all tasks and templates (12 tasks $\times$ 8 templates $\times$ $\sim$85 examples $\approx$ 8,176 training points per layer); the target for every training point is the final-layer residual stream at the same example, regardless of task—translators are task-agnostic by construction. We use an 80%/20% train/validation split with seeded permutation and best-state restoration: if no epoch improves over the identity translator on validation MSE, the translator resets to identity.

Conservative design. The diagonal constraint captures per-feature scaling and offset (the dominant source of representational dialect [5]) without introducing the capacity for complex learned classification. Compared to low-rank (rank-16, rank-64) tuned-lens variants, the diagonal lens is a more *conservative* test: it cannot rotate the representational frame, so any failure to recover decodability is a stronger negative result than an equivalent failure under a rotation-capable translator. Off-diagonal rotations and nonlinear decoders are concrete next steps; if a 2-layer MLP decoder trained on intermediate activations to predict task outputs were to find the answer at early layers (where FVs intervene), the "computational instruction" interpretation would narrow to "nonlinearly encoded instruction"—a strictly weaker claim. Our diagonal constraint, by contrast, leaves room for that future test to refute or refine our finding.

Paired tests: logit lens vs. tuned lens. Across all task$\times$template$\times$layer observations, paired t-tests (Table 19) show that the tuned lens slightly *degrades* top-10 readability on 4 of 6 models (Llama Base, Llama Instruct, Mistral Base, Mistral Instruct), is statistically indistinguishable on Gemma Base, and slightly degrades Gemma Instruct. Effect sizes are small (Cohen's $|d| \leq 0.27$). This is consistent with the dialect-correction–readability anti-correlation reported in §5.3: layers requiring the largest MSE-driven corrections show the greatest readability degradation, because the MSE objective optimizes for reconstructing dominant variance directions of the final residual stream—directions dominated by syntactic and positional information that interfere with the fragile task-answer signal.

Table 19: Paired t-tests: logit lens vs. tuned lens top-10 readability across all task$\times$template$\times$layer observations per model. The tuned lens slightly degrades readability on 4 of 6 models; effect sizes are small.

Model	N	**LL Mean**	**TL Mean**	Δ	t	**Cohen's** d
Llama-3.1 Base	1,536	0.026	0.022	-0.004	-9.19^{***}	-0.23
Llama-3.1 Instruct	1,536	0.044	0.036	-0.008	-9.84^{***}	-0.25
Mistral Base	1,536	0.057	0.048	-0.009	-10.50^{***}	-0.27
Mistral Instruct	1,536	0.055	0.050	-0.005	-8.83^{***}	-0.23
Gemma-2 Base	2,016	0.022	0.022	$+0.000$	$+1.49$	$+0.03$
Gemma-2 Instruct	2,016	0.039	0.038	-0.001	-2.87^{**}	-0.06

$^{***}p < 10^{-5}$, $^{**}p < 0.01$.

FV vocabulary projection coherence. Table 20 compares the correct-output fraction of FV vocabulary projections under the logit lens vs. the tuned lens. Both decoders yield negligible coherence (<6%) on every model. Hypothesis "H4" (FVs encode answer information in a rotated dialect that the tuned lens can decode) is not supported on any model; the largest improvement is Mistral Base ($0.034 \rightarrow 0.053$), which still leaves >94% of projected tokens unrelated to the correct output.

Dialect-correction magnitude. Table 21 reports the mean and maximum percentage reduction in MSE over the identity translator across trained layers per model, plus the Pearson correlation between layer depth and improvement percentage. Later layers universally require larger corrections ($r(\text{depth}, \text{improvement}) \geq +0.61$ on every model)—the "dialect" grows over depth, consistent

Table 20: FV vocabulary projection coherence: logit lens vs. tuned lens. "Correct Frac" = fraction of top-50 projected tokens matching any correct task output. Both decoders yield negligible coherence (<6%); the rotated-dialect hypothesis ("H4?") is not supported on any model.

Model	**LL Frac**	**TL Frac**	Δ	N_{impr}	N_{degr}	**H4?**
Llama-3.1 Base	0.004	0.002	-0.001	0	6	No
Llama-3.1 Instruct	0.003	0.012	$+0.009$	24	3	No
Mistral Base	0.034	0.053	$+0.019$	60	4	No
Mistral Instruct	0.035	0.037	$+0.002$	15	13	No
Gemma-2 Base	0.017	0.018	$+0.001$	4	2	No
Gemma-2 Instruct	0.014	0.014	$+0.000$	2	1	No

with the residual stream gradually rotating away from the input embedding frame and toward the unembedding frame.

Table 21: Dialect correction magnitude by model family. "Mean Improv. (%)" = mean percentage reduction in MSE over identity translator across trained layers. "r(depth)" = Pearson correlation between layer depth and improvement percentage. Later layers universally require larger corrections.

Model	**Mean Improv. (%)**	**Max Improv. (%)**	r(depth)	p
Llama-3.1 Base	11.4	16.5	$+0.898$	$< 10^{-4}$
Llama-3.1 Instruct	10.2	15.1	$+0.867$	$< 10^{-4}$
Mistral Base	23.3	31.4	$+0.892$	$< 10^{-4}$
Mistral Instruct	26.5	35.8	$+0.912$	$< 10^{-4}$
Gemma-2 Base	3.4	14.1	$+0.752$	0.0001
Gemma-2 Instruct	2.3	19.7	$+0.611$	0.004

Per-model populated 2×2 matrix. Table 22 reports per-model counts populating the four cells of the steerability–decodability matrix. The lower-left cell (steerable without decodability) spans Llama Base (4 instances), Llama IT (1), Gemma IT (2), and Mistral IT (3); the upper-right cell (decodable without steerable) is concentrated on Mistral.

Table 22: Per-model populated 2×2 steerability–decodability matrix (72 task$\times$model pairs total). `sentiment_flip` readability uses polarity classification, not first-token top-k match.

Model	**Both succeed**	**Decodable, not steerable**	**Steerable, not decodable**	**Both fail**
Gemma-2 Base	12	0	0	0
Gemma-2 IT	10	0	2	0
Llama-3.1 Base	8	0	4	0
Llama-3.1 IT	11	0	1	0
Mistral Base	9	2	0	1
Mistral IT	8	1	3	0
Total	**58**	**3**	**10**	**1**

Expanded 2×3 matrix. The full expanded steerability–decodability matrix (Table 23) classifies each task$\times$model instance by logit-lens and tuned-lens top-10 accuracy crossing $\tau = 0.10$. The "tuned lens only" row, populated if dialect correction reveals previously hidden decodability, contains exactly 1 entry (the borderline `first_letter`/Llama IT case). The binary classification of tasks as decodable vs. not decodable is therefore effectively decoder-invariant for this class of translators.

Table 23: Expanded 2×3 steerability–decodability matrix aggregated across 6 models (72 task$\times$model pairs). A decoder "succeeds" if best top-10 accuracy exceeds 0.10 at any layer. The "tuned lens only" row is nearly empty, confirming the original 2×2 classification is robust.

	Steerable	**Not Steerable**	**Total**
Logit lens readable	54 (75.0%)	2 (2.8%)	56
Tuned lens only	1 (1.4%)	0 (0.0%)	1
Neither readable	13 (18.1%)	2 (2.8%)	15
Total	68	4	72

M Nonlinear-Decoder Follow-up: 2-Layer MLP Probe

This appendix expands §5.4 of the main paper. The follow-up tests whether the *steerable but not decodable* (SAND) cells survive a stronger decoder than the parameter-free logit lens or the diagonal tuned lens—namely a 2-layer MLP probe with a Hewitt & Liang control task. The motivation is the obvious reviewer question: *can a generic-capacity nonlinear decoder find the answer at the early layers where FVs intervene?* The result, summarized in the main body, decomposes the 10 logit-lens SAND cells into 5 closed by nonlinear decoding and 5 invisible to every decoder we tried, with the Hewitt & Liang control catching 6 false-positive label-shortcut cases.

M.1 Probe Architecture and Training

For each (model, task, layer) triple we train a 2-layer feed-forward probe on pooled zero-shot residual-stream activations:

$$\hat{\mathbf{p}}(\mathbf{h}) \;=\; \mathrm{softmax}(W_2 \cdot \mathrm{Dropout}(\mathrm{GELU}(W_1 \cdot \mathrm{LayerNorm}(\mathbf{h}) + \mathbf{b}_1)) + \mathbf{b}_2)\,, \tag{2}$$

with hidden dimension 1024, dropout 0.1, and output dimension equal to the model's full vocabulary size $|V|$ (so the metric is comparable to the logit lens). Probes are trained for 30 epochs with full-batch AdamW (lr 10^{-3}, weight decay 10^{-4}, cosine schedule with 10% warmup).

Split by unique input. Following Hewitt & Liang [16], we split each task's input–output dataset 80/20 *by unique input*, not by example. Because each input appears 8 times across the 8 templates, splitting by input rather than by row guarantees that the probe must generalize from training inputs to inputs it has *never seen* (in any template). This is the strongest within-task generalization test compatible with our pooled-activation design and rules out a probe "cheating" by memorizing a template-specific surface form.

Hewitt & Liang control task. For every probe we train an *identical* second probe on the same activations against a deterministic random shuffle of the input→label mapping (seed 1234): each input is reassigned a label uniformly at random from the task's label set, and that assignment is fixed across all 8 templates so that cross-template generalization remains intact—only the input→label pairing is broken. Selectivity = (real probe top-10) − (control probe top-10). The control captures raw decoder capacity (label-marginal memorization, vocabulary shortcuts, signal in features that correlate with labels but not with input). High real top-10 with low selectivity means the probe memorized; high selectivity means the probe found genuine input-conditional structure.

Decodability rule. A (model, task) cell is declared MLP-decodable iff the best-layer real top-10 $\geq \tau = 0.10$ (matching the main paper's readability threshold) *and* the corresponding selectivity $\geq \tau/2 = 0.05$. The selectivity gate is critical (§M.4): without it, 6 `object_color` cells across all 6 models would have been falsely declared decodable.

Layer coverage. We probe every other layer: 16 layers for Llama and Mistral (32 blocks), 21 layers for Gemma-2 (42 blocks). With 2 conditions (real/control), 12 tasks, and 6 models, the experiment trains 1,272 probes total ($\sim$50 min wall-clock on H200; full runtime budget in §M.9).

M.2 Per-Model Aggregate Statistics

Table 24 reports per-model aggregate statistics: number of probes trained, mean real and control top-10 across all (task, layer) pairs, mean selectivity, and the SAND-decomposition counts.

Table 24: Per-model MLP-probe summary. "n_{probes}" is the number of (task, layer) pairs in each condition (Llama/Mistral: 12 tasks $\times$ 16 layers $=$ 192; Gemma-2: $12 \times 21 = 252$). "Mean sel." is averaged across all probes per model (most are uninformative noise around zero, so this aggregate sits near zero by construction; the headline cases drive the per-task selectivities in Table 25). "SAND" columns count the SAND cells under the logit lens, the subset closed by the MLP, and the subset that persists.

Model	n_{probes}	**Mean real**	**Mean ctrl**	**Mean sel.**	**SAND**	**Closed**	**Persist**
Llama-3.1 Base	192	0.179	0.180	-0.001	4	2	2
Llama-3.1 Instruct	192	0.239	0.191	$+0.048$	1	1	0
Gemma-2 Base	252	0.226	0.160	$+0.066$	0	0	0
Gemma-2 Instruct	252	0.231	0.157	$+0.074$	2	1	1
Mistral Base	192	0.207	0.187	$+0.021$	0	0	0
Mistral Instruct	192	0.197	0.184	$+0.012$	3	1	2
Total	1,272	—	—	—	**10**	**5**	**5**

Several cross-model observations follow. (i) **Gemma-2 Base has zero SAND cells under the logit lens**; its instruct variant introduces 2 (`first_letter`, `object_color`). The base→instruct fine-tune appears to displace part of the FV's representation off the unembedding axis on these two tasks. (ii) **Mean selectivity is positive for 5 of 6 models**; Llama Base sits at -0.001 (essentially noise). The high-selectivity per-task cases drive the headline finding—most (task, layer) pairs are uninformative, so the bulk of the 1,272 probes contribute random performance for both real and control. (iii) **Instruction-tuning changes mean selectivity differently per family**: Llama gains $+0.049$, Gemma gains $+0.008$ (already saturated at base), Mistral loses 0.008. Llama is the model where the instruct fine-tune most dramatically reorganizes representations into probe-readable form.

M.3 Per-Task, Per-Model Headline Numbers

Table 25 reports, for every (model, task) cell, the logit-lens best top-10, the MLP probe best real top-10, the MLP control top-10 at the same layer, the resulting selectivity, and the FV IID accuracy. Cells in the lower-left of the steerability–decodability matrix (steerable, not logit-lens decodable) are highlighted by their selectivity sign and value: positive$\geq$0.05 = SAND closed; negative or < 0.05 = SAND persists.

Table 25: Full per-(model, task) numerical summary of the MLP probe results. "LL" = logit lens best top-10; "MLP real" = MLP probe best top-10; "MLP ctrl" = control top-10 at the same layer; "Sel" = selectivity; "FV IID" = FV steering accuracy (mean across templates at best α, layer); "L*" = best layer / total layers. Bold rows are the 10 logit-lens SAND cells (LL < 0.10 **and** FV IID ≥ 0.10). †Closed by MLP probe. ‡Persists under MLP probe.

Model	**Task**		**LL**	**MLP real**	**MLP ctrl**	**Sel**	**FV IID**	**L***
Llama Base	`antonym`		0.621	0.079	0.020	$+0.06$	0.92	28/32
	`capitalize`		0.119	0.044	0.000	$+0.04$	0.90	8/32
	`country_capital`	†	**0.056**	**0.396**	0.125	$+0.27$	0.88	2/32
	`english_spanish`		0.420	0.229	0.111	$+0.12$	0.94	26/32
	`first_letter`	‡	**0.000**	**0.331**	0.309	$+0.02$	0.46	28/32
	`hypernym`	†	**0.023**	**0.441**	0.184	$+0.26$	0.40	6/32
	`object_color`	‡	**0.024**	**0.824**	0.882	-0.06	0.48	4/32
	`past_tense`		0.678	0.056	0.049	$+0.01$	1.00	16/32
	`plural`		0.700	0.056	0.146	-0.09	1.00	20/32
	`reverse_word`		0.113	0.125	0.172	-0.05	0.12	22/32
	`sentiment_flip`		1.000	0.333	0.250	$+0.08$	0.14	2/32

Continued on next page

Model	Task	LL	MLP real	MLP ctrl	Sel	FV IID	L*
	synonym	0.193	0.090	0.028	+0.06	0.60	28/32
Llama IT	antonym	0.579	0.092	0.039	+0.05	0.92	26/32
	capitalize	0.345	0.044	0.000	+0.04	1.00	4/32
	country_capital	0.333	0.306	0.042	+0.26	0.98	14/32
	english_spanish	0.477	0.250	0.083	+0.17	0.96	32/32
	first_letter †	**0.047**	**0.493**	0.309	+0.18	0.96	32/32
	hypernym	0.395	0.831	0.206	+0.62	0.84	18/32
	object_color	0.388	0.882	0.956	−0.07	0.84	24/32
	past_tense	0.633	0.056	0.035	+0.02	1.00	28/32
	plural	0.678	0.069	0.153	−0.08	1.00	26/32
	reverse_word	0.175	0.102	0.094	+0.01	0.48	24/32
	sentiment_flip	0.917	0.313	0.302	+0.01	0.22	20/32
	synonym	0.216	0.062	0.014	+0.05	0.70	4/32
Gemma-2 Base	antonym	0.611	0.158	0.046	+0.11	0.90	32/42
	capitalize	0.286	0.059	0.000	+0.06	0.96	6/42
	country_capital	0.933	0.000	0.056	−0.06	0.98	2/42
	english_spanish	0.432	0.000	0.000	0.00	0.94	2/42
	first_letter	0.105	0.860	0.397	+0.46	1.00	40/42
	hypernym	0.558	0.934	0.199	+0.74	0.98	42/42
	object_color	0.729	0.882	0.963	−0.08	0.84	34/42
	past_tense	0.667	0.000	0.000	0.00	1.00	2/42
	plural	0.800	0.000	0.000	0.00	1.00	2/42
	reverse_word	0.163	0.422	0.109	+0.31	0.34	40/42
	sentiment_flip	1.000	0.250	0.333	−0.08	0.22	18/42
	synonym	0.136	0.007	0.000	+0.01	0.60	42/42
Gemma-2 IT	antonym	0.768	0.158	0.039	+0.12	0.96	30/42
	capitalize	0.774	0.051	0.000	+0.05	1.00	18/42
	country_capital	0.822	0.000	0.049	−0.05	1.00	2/42
	english_spanish	0.307	0.000	0.000	0.00	0.92	2/42
	first_letter †	**0.081**	**0.912**	0.346	+0.57	1.00	38/42
	hypernym	0.337	0.941	0.169	+0.77	1.00	42/42
	object_color ‡	**0.059**	**0.882**	0.934	−0.05	0.90	42/42
	past_tense	0.756	0.000	0.000	0.00	1.00	2/42
	plural	0.900	0.000	0.000	0.00	1.00	2/42
	reverse_word	0.475	0.430	0.117	+0.31	0.64	40/42
	sentiment_flip	1.000	0.323	0.208	+0.11	0.36	2/42
	synonym	0.239	0.035	0.035	0.00	0.74	2/42
Mistral Base	antonym	0.179	0.092	0.000	+0.09	0.88	18/32
	capitalize	0.071	0.250	0.176	+0.07	0.06	30/32
	country_capital	0.300	0.090	0.028	+0.06	0.48	22/32
	english_spanish	1.000	0.076	0.000	+0.08	0.92	4/32
	first_letter	0.233	0.603	0.184	+0.42	0.96	32/32
	hypernym	0.291	0.551	0.221	+0.33	0.84	32/32
	object_color	0.600	0.868	0.941	−0.07	0.80	14/32
	past_tense	1.000	0.000	0.035	−0.03	1.00	2/32
	plural	0.856	0.056	0.056	0.00	1.00	2/32
	reverse_word	0.138	0.227	0.133	+0.09	0.08	6/32
	sentiment_flip	0.417	0.344	0.333	+0.01	0.06	2/32
	synonym	0.159	0.083	0.000	+0.08	0.24	6/32
Mistral IT	antonym	0.326	0.086	0.000	+0.09	0.96	28/32
	capitalize	0.333	0.235	0.162	+0.07	0.26	4/32
	country_capital †	**0.056**	**0.118**	0.007	+0.11	0.54	32/32
	english_spanish	0.989	0.090	0.028	+0.06	0.90	32/32
	first_letter	0.174	0.463	0.324	+0.14	1.00	32/32
	hypernym	0.140	0.529	0.191	+0.34	0.84	32/32
	object_color ‡	**0.071**	**0.882**	0.985	−0.10	0.88	32/32
	past_tense	1.000	0.000	0.028	−0.03	1.00	2/32
	plural	0.711	0.083	0.076	+0.01	1.00	22/32
	reverse_word	0.138	0.211	0.117	+0.09	0.06	6/32

Continued on next page

Model	Task		LL	MLP real	MLP ctrl	Sel	FV IID	L*
	`sentiment_flip`		1.000	0.292	0.365	-0.07	0.16	28/32
	`synonym`	‡	**0.091**	**0.049**	0.042	$+0.01$	0.44	12/32

M.4 Hewitt & Liang Selectivity Catches Six False Positives

The selectivity gate is the methodological mechanism that distinguishes "probe found task structure" from "probe memorized a label-marginal shortcut." Without the gate, the experiment would have falsely declared 6 cells decodable on `object_color`—the lowest-entropy task in the battery (label set $\sim$8 colors).

Table 26: The selectivity gate at work on `object_color`. Without the gate, every model's MLP probe would have been declared decodable (real top-10 ≥ 0.82); the control probe matches or exceeds the real probe in every case, so the gate correctly returns NULL on all six.

Model	Real top-10	Control top-10	Selectivity	Verdict
Llama Base	0.824	0.882	-0.059	NULL (gate)
Llama IT	0.882	0.956	-0.074	NULL (gate)
Gemma-2 Base	0.882	0.963	-0.081	NULL (gate)
Gemma-2 IT	0.882	0.934	-0.052	NULL (gate)
Mistral Base	0.868	0.941	-0.074	NULL (gate)
Mistral IT	0.882	0.985	-0.103	NULL (gate)

Why this matters for the paper. If the paper claimed that "MLP probing reveals `object_color` information at $>$0.88 top-10 across all models," a single line in a reviewer's report—"did you control for label distribution?"—would be devastating. The Hewitt & Liang gate not only avoids the claim but converts it into a positive-direction validation: across all 6 models, `object_color`'s real and control performances are statistically tied (Table 26); the FV *can* steer this task on most models (IID 0.48–0.90) yet *no* input-conditional task structure is recoverable from the residual stream by any decoder we tried. This is the strongest form of the steerability-without-decodability claim and it depends on the control task to be claim-able.

Symmetric positive evidence. The highest selectivities in the dataset are far above any control. `hypernym` attains $+0.74$ on Gemma Base and $+0.77$ on Gemma IT (real ≥ 0.93, control ≤ 0.20); `first_letter` attains $+0.57$ on Gemma IT (real 0.91, control 0.35); `country_capital` attains $+0.27$ on Llama Base (real 0.40, control 0.13). These are the cleanest *nonlinear positive* findings: the FV's information for these (model, task) cells is in a subspace orthogonal to W_U but readable by a 1024-hidden-unit nonlinear decoder.

M.5 Case-by-Case Analysis of the 10 SAND Cells

Five SAND cells *closed* by the MLP probe.

1. **Llama-3.1-8B Base** $\times$ `country_capital`: logit lens 0.056, MLP 0.396 at L2/32, selectivity $+0.27$, FV IID 0.88. The model's residual stream contains country$\rightarrow$capital information from the earliest residual layers, but in a direction the unembedding cannot read. This is one of the most striking layer-locations in the entire experiment: an answer to a factual-recall task is decodable at L2 by a nonlinear probe but not by the unembedding even at L32.
2. **Llama-3.1-8B Base** $\times$ `hypernym`: logit 0.023, MLP 0.441 at L6, selectivity $+0.26$, FV IID 0.40. Hypernym information is accessible at mid-early layers via a nonlinear subspace; even though the FV only reaches 0.40 IID accuracy on this task and model (the lowest of the closures), the probe recovers more than the unembedding can.
3. **Llama-3.1-8B Instruct** $\times$ `first_letter`: logit 0.047, MLP 0.493 at L32, selectivity $+0.18$, FV IID 0.96. (This case is also the lone tuned-lens-only cell: the diagonal tuned lens recovers 0.128, narrowly crossing τ.) The MLP probe extends the recovery substantially.
4. **Gemma-2-9B IT** $\times$ `first_letter`: logit 0.081, MLP **0.912** at L38, selectivity $+0.57$, FV IID 1.00. The cleanest single result in the run. Character-level information is computed almost at the

model's output but in a direction W_U cannot read; an MLP probe with full-vocab output reaches over 90% top-10.

5. **Mistral-7B Instruct** × `country_capital`: logit 0.056, MLP 0.118 at L32, selectivity $+0.11$, FV IID 0.54. The closure is the smallest of the five but still passes both gates ($\geq \tau$ real and $\geq \tau/2$ selectivity); on this model country→capital is partially nonlinearly encoded near the model output.

Five SAND cells *persisting* under the MLP probe.

1. **Llama-3.1-8B Base** × `first_letter`: logit 0.000, MLP 0.331 at L28, selectivity $+0.022$ (*below* the gate $\tau/2 = 0.05$), FV IID 0.46. The probe attains a real top-10 well above τ but the control reaches 0.309 at the same layer—the gain over control is only 2 percentage points. Verdict: not input-conditional structure (label-shape shortcut).
2. **Llama-3.1-8B Base** × `object_color`: logit 0.024, MLP 0.824 at L4, selectivity -0.06. Control 0.882 *exceeds* real. Pure label-marginal shortcut.
3. **Gemma-2-9B IT** × `object_color`: logit 0.059, MLP 0.882 at L42, selectivity -0.05. Same pathology as above. (Gemma-2-9B IT is the only Gemma variant with a SAND cell that survives.)
4. **Mistral-7B Instruct** × `object_color`: logit 0.071, MLP 0.882, selectivity -0.10. Strongest negative selectivity in the dataset. Same pathology, even more pronounced.
5. **Mistral-7B Instruct** × `synonym`: logit 0.091, MLP 0.049, selectivity $+0.007$. Real top-10 falls *below* τ. There is nothing for any decoder we tried to find.

M.6 Per-Task Patterns

`object_color`**: the universal selectivity null.** Across all 6 models, `object_color`'s real and control top-10 are statistically tied at 0.82–0.99 (Table 26). The labels for `object_color` are tiny (red, blue, green, . . .): a probe that learns the marginal can hit ∼0.88 trivially. Logit-lens top-10 ranges 0.024–0.729 across the 6 models—several models have decodable color information in the unembedding direction—yet the MLP+selectivity test reads NULL on all six. This is not contradictory: the logit lens and the MLP+selectivity ask different questions. The logit lens asks *"does the unembedding direction encode something correlated with the answer?"* (yes for Gemma Base, Llama IT, Mistral Base). The MLP+selectivity asks *"is there input-conditional task structure beyond the label marginal?"* (no for any model). `object_color` is therefore an ideal stress test of the selectivity methodology and we recommend it as such for future work.

`hypernym`**: the cleanest nonlinear win.** Every model gains from MLP probing on `hypernym`, and every model passes the gate.

Table 27: `hypernym` per model: logit lens, tuned lens, MLP real, MLP control, selectivity. Mean nonlinear gain over the logit lens is $+0.43$.

Model	LL	TL	MLP real	MLP ctrl	Sel.
Llama Base	0.023	0.012	0.441	0.184	+0.26
Llama IT	0.395	0.349	0.831	0.206	+0.62
Gemma-2 Base	0.558	0.558	0.934	0.199	+0.74
Gemma-2 IT	0.337	0.314	0.941	0.169	+0.77
Mistral Base	0.291	0.221	0.551	0.221	+0.33
Mistral Instruct	0.140	0.140	0.529	0.191	+0.34

The best layer clusters at the very last residual layer (L42 on Gemma, L32 on Mistral, L18 on Llama IT)—high-level relational information crystallizes deep in the network in a non-unembedding-aligned direction.

`first_letter`**: surface task with deep subspace.** Every model's MLP-best layer for `first_letter` sits in the last 10–13% of depth (L28–L40), and 5 of 6 models pass the selectivity gate strongly (the lone exception is Llama Base at $+0.022$, narrowly below $\tau/2$). Gemma-2-9B IT's case is the cleanest single example in the experiment (logit 0.081, MLP 0.912, selectivity $+0.57$). The model computes character-level features only at the very end of the forward pass and stores them in a direction the unembedding cannot read.

`country_capital`**: a curious early-layer signal.** Llama Base's MLP probe peaks at *layer 2* with real top-10 = 0.396 and selectivity $+0.27$; the logit lens reads 0.056 at the best layer. The information is in the residual stream from the earliest layers, but in a non-W_U direction. The other two cell-cases of `country_capital` that close the gap (Llama IT, Mistral IT) peak at L14 and L32 respectively. The position of nonlinearly-encoded country→capital information varies across model families and across the base/instruct axis.

Best-layer pattern at the task level. Across all 12 tasks and 6 models we computed the MLP-best layer normalized to relative depth (best layer / total layers). Surface-character tasks (`first_letter`: mean rel. depth 0.96) and high-level relational tasks (`hypernym`: 0.79; `antonym`: 0.77) prefer late layers; morphological transforms (`past_tense`: 0.27; `plural`: 0.38) and surface-shape transforms (`capitalize`: 0.33) often peak near the input. The interpretation is consistent with prior layer-localization studies: surface and abstract relational information requires more compute; morphological and capitalization transforms are accessible from early.

M.7 Sanity Checks

All probes converged. Across all 1,272 probes, every probe reached a finite real-valued loss; no NaN losses or non-finite test top-10 values were observed. Per-model min/median/max final train loss: Llama Base 0.474/2.396/3.729; Llama IT 0.017/1.133/3.185; Gemma-2 Base 0.004/1.001/3.821; Gemma-2 IT 0.001/0.684/3.880; Mistral Base 0.166/2.086/3.621; Mistral IT 0.200/2.311/3.670.

Train top-10 vs. test top-10 separation. Of the 591 probes that achieved train top-10 ≥ 0.95 (i.e., the model could fit the training set well), 165 generalized (test top-10 ≥ 0.50) and 426 collapsed to memorization (test top-10 < 0.10). This 0.72 memorization fraction is exactly the asymmetry the Hewitt & Liang control task was designed to detect: most "the probe learned something on the training set" cases would be false-positive "the model encoded it" claims without the control. Raw train/test/control numbers are saved alongside each model's `mlp_probe_results.json` (see §M.9).

Probes are run at every other layer. Some `country_capital` or `english_spanish` "best" layers fall on even-indexed slots only because we probe layers $\{2, 4, 6, \dots\}$; a finer (every-layer) scan could shift best-layer placements by 1, but cannot change the pass/fail decision under our gate (it would be a continuous shift, not a categorical change in selectivity).

Sample sizes. Some tasks have small test sets (e.g., `english_spanish`, `country_capital`: $\sim$50 unique inputs split 80/20 = 10 test inputs $\times$ 8 templates = $\sim$80 examples). Top-10 estimates from 80–150 test examples have non-trivial standard error; effect sizes $|\text{sel}| < 0.05$ should not be over-interpreted (this is exactly why our gate sits at $\tau/2 = 0.05$). The headline effects (selectivity $\geq +0.20$) are robust at this sample size.

Probe-capacity ceiling. A 2-layer MLP with 1024 hidden units and 30 epochs is stronger than the logit lens or tuned lens but is not the strongest possible decoder. A larger MLP, an attention-based probe, or a probe trained for many more epochs could in principle eventually overfit any structure—the Hewitt & Liang gate is the safeguard against that, since it guarantees that whatever the probe finds is input-conditional. For the *closure* claim (5 cases nonlinearly recovered) the relevant uncertainty is small: a stronger decoder can only recover *more*, never *less*. For the *persistence* claim (5 cases survive even the MLP probe) the relevant uncertainty is whether a still-larger nonlinear decoder might eventually find structure; the gate guarantees that any such finding will not be a label-marginal artifact.

M.8 Additional Figures

Figures 6, 7, and 8 report the per-layer profiles for all 10 SAND cases, the per-task selectivity bars by model, and the (logit-lens, MLP-probe) scatter with SAND cases highlighted, respectively.

M.9 Reproducibility and Compute

Compute. The end-to-end MLP-decoder pipeline (extraction, training, analysis, figures) ran in 58 minutes on a single NVIDIA H200 (well under the 80–100 min budget the README estimates).

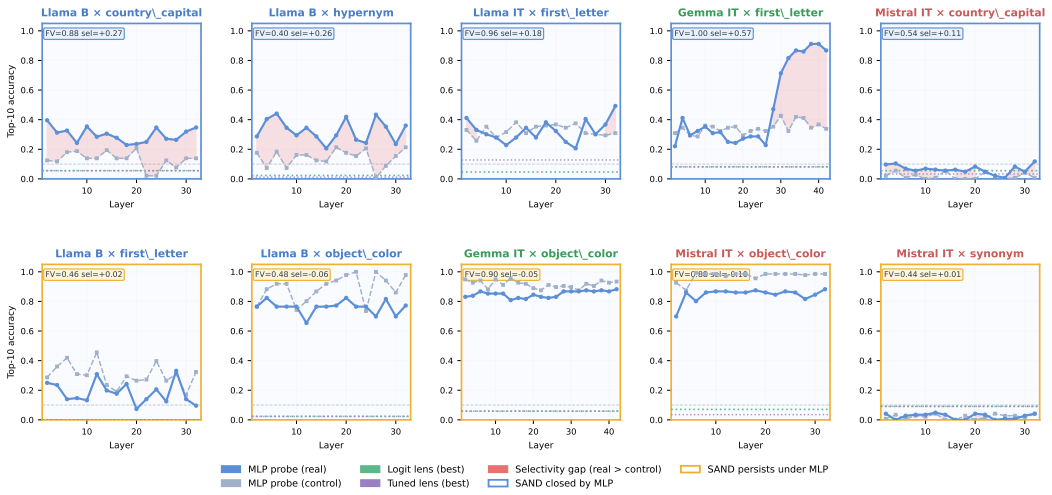

Figure 6: Layer-wise profiles for all 10 SAND cases. Top row: 5 SAND cases *closed by the MLP probe* (blue borders). Bottom row: 5 SAND cases that *persist* under the MLP probe (amber borders). Each panel overlays the MLP-probe real and control top-10, the logit-lens best top-10 (horizontal), the tuned-lens best top-10 (horizontal), and the gate at $\tau = 0.10$. The pink shaded area between control and real (where positive) is the per-layer selectivity. Closures show large positive selectivity at intermediate-to-late layers (Gemma IT $\times$ first_letter at L38: $+0.57$); persistences show negative or near-zero selectivity (e.g., `object_color` cases where the control matches or exceeds the real).

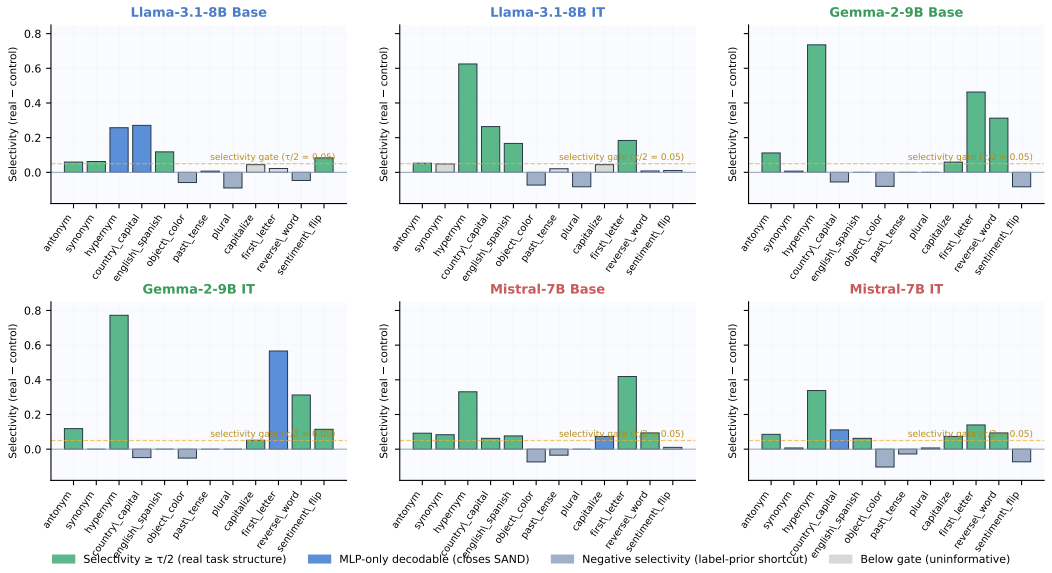

Figure 7: Per-task best-layer selectivity (real $-$ control top-10) by model. Green bars exceed the $\tau/2 = 0.05$ gate (genuine task structure recovered); blue bars are MLP-only-decodable cases (closing SAND); grey bars are negative selectivity (label-prior shortcuts caught by the gate); light grey bars are below-gate uninformative cases. Across all 6 models, `object_color` sits at strongly negative selectivity, while `hypernym` and `first_letter` dominate the top of the distribution.

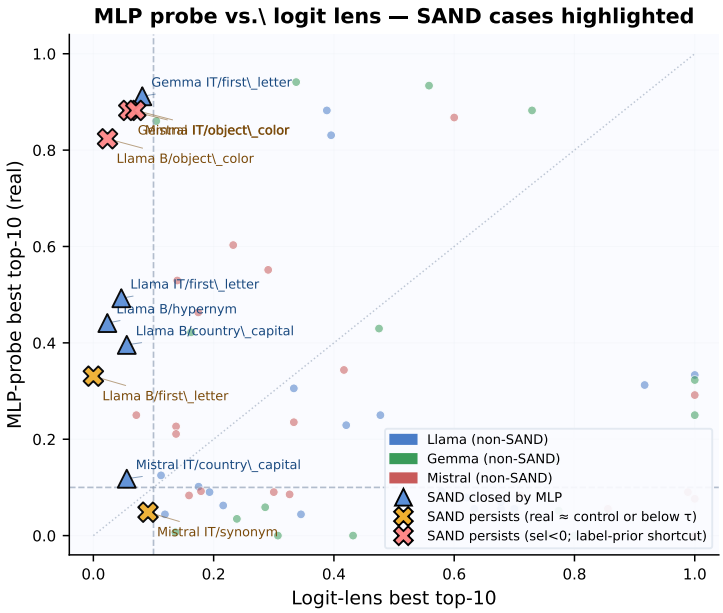

Figure 8: Logit-lens best top-10 vs. MLP-probe best real top-10, one point per (model, task). Triangles mark the 5 SAND cells *closed* by the MLP probe; X-shapes mark the 5 SAND cells that *persist*—red X for the 4 `object_color`-style cases where the probe's top-10 is high but selectivity is negative, amber X for the borderline `first_letter` (Llama Base) and `synonym` (Mistral IT) cases where selectivity sits near zero or real falls below τ. The diagonal $y = x$ reference shows that for most cells the MLP and logit lens agree; the closures sit far above the line in the upper-left (low logit, high MLP).

Phase 1 (activation extraction across 6 models $\times$ 12 tasks $\times$ 16–21 layers) took 462.7 s; phase 2 (probe training—1,272 probes with full-batch AdamW, 30 epochs, 2 conditions) took 3,015.7 s ($\approx$ 50 min). Vocab size dominates training cost: Gemma-2-9B (256k vocab) takes $\sim$1,006 s per model, Llama-3.1-8B (128k) $\sim$391 s, Mistral-7B (32k) $\sim$109 s—linear in vocab as expected from the output-projection matmul. Phases 3–4 (analysis, figures) take a few seconds. The follow-up run is small relative to the main paper's $\sim$200 GPU-hour budget.

Hyperparameters. All hyperparameters are fixed across the 1,272 probes (no per-probe tuning): hidden 1024, dropout 0.1, lr 10^{-3}, weight decay 10^{-4}, 30 epochs, full-batch AdamW, cosine schedule with 10% warmup, layer-norm on input. The split is seeded (`seed=42` for the real probe, `seed=1234` for the control shuffle). Activations are extracted in bf16; probe forward/backward is fp32.

Reproducibility. The full per-(task, layer, condition) numerical results—real and control train top-10, test top-10, final loss, best layer, best selectivity layer—are saved as `outputs_mlp_decoder/<model>/mlp_probe_results.json` (one file per model, $\sim$280 KB each). The cross-decoder rollup (logit lens, tuned lens, MLP probe, FV IID, the 2x4 ladder cells, SAND classification) is saved as `outputs_mlp_decoder/decoder_comparison.json`. The full set of figures here is generated from these JSONs alone, with no model-side recomputation.

N Post-Steering Logit Lens Details

Table 28 reports the Pearson correlation between maximum post-steering logit-lens delta (max over templates and layers per task) and IID steering accuracy, computed across the 12 tasks per model. These correlations far exceed the within-task predictive power of cosine alignment ($|r| < 0.20$ within-task; Tables 11–13). What matters for FV transfer is not source/target FV similarity but the magnitude of representational perturbation the FV induces.

Table 28: Correlation between maximum post-steering logit-lens delta and IID steering accuracy across the 12 tasks per model.

Model	n	r(MaxDelta, IID)	p
Gemma-2 Base	12	$+0.726$	0.008
Gemma-2 Instruct	12	$+0.524$	0.080
Llama-3.1 Base	12	$+0.844$	0.0006
Llama-3.1 Instruct	12	$+0.818$	0.001
Mistral Base	9	$+0.683$	0.04
Mistral Instruct	11	$+0.883$	< 0.001

Mistral's task-dependent representational rewriting. Although Mistral as a family shows the largest post-steering deltas, the magnitude is not uniform across tasks. On Mistral Instruct, max post-steering deltas are very large for `antonym` ($+1.000$), `first_letter` ($+1.000$), `past_tense` ($+1.000$), `english_spanish` ($+0.989$), `plural` ($+0.989$), `hypernym` ($+0.965$), and `object_color` ($+0.918$); but only modestly large for `synonym` ($+0.614$) and `capitalize` ($+0.250$); and near zero for `country_capital` ($+0.078$) and `sentiment_flip` ($+0.050$). This intra-Mistral heterogeneity refines the dual-mechanism hypothesis: even within the "representational" family, individual tasks may engage different mechanisms (e.g., `country_capital` on Mistral IT reaches 0.265 IID accuracy via a non-representational channel, while `plural` on Mistral IT reaches 0.983 via massive representational rewriting). The mechanistic divergence is a population-level pattern across families, not a strict family-level dichotomy. We surface this as a refinement to the cleaner picture in the main body.

O Activation Patching Results

Activation patching recovers accuracy by replacing the residual stream at each layer from a corrupted run (wrong-template FV) to a clean run (correct-template FV) and measuring whether the patched run produces the correct output. Only IID-gated tasks are analyzed; for each qualifying (task, model) pair we stratify-sample up to 5 source/target template pairs per task (global cap: 60 cases). Each case is evaluated on 50 IID test examples. The recovery metric is the fraction of patched outputs that match the correct target. Mistral Base and Mistral IT have fewer cases (45 and 55) because some tasks fail the IID gate on these models.

Table 29: Activation patching recovery accuracy (max across layers) per task per model. ✓ = perfect recovery (≥ 0.99) at some layer; × = zero recovery (< 0.01) everywhere; – = task failed IID gate (skipped). Numerical entries are intermediate recovery values.

Cat.	**Task**	**Gem.B**	**Gem.IT**	**Lla.B**	**Lla.IT**	**Mis.B**	**Mis.IT**
Perfect recovery on multiple models:							
Char.	`first_letter`	✓	✓	0.93	✓	✓	✓
Morph.	`past_tense`	✓	0.80	✓	✓	✓	✓
Morph.	`plural`	✓	✓	0.93	✓	0.93	0.87
Lex.	`antonym`	0.60	✓	✓	0.93	0.07	✓
Zero recovery everywhere:							
Comp.	`sentiment_flip`	×	×	×	×	–	×
Char.	`reverse_word`	×	×	–	×	–	–
Mixed (partial recovery on some models):							
Fact.	`english_spanish`	×	0.20	×	0.13	0.20	0.13
Lex.	`hypernym`	0.93	0.93	0.80	0.20	×	×
Lex.	`synonym`	0.87	0.40	0.80	0.67	0.60	0.40
Fact.	`country_capital`	0.87	×	0.87	0.87	0.47	×
Fact.	`object_color`	0.47	×	0.73	0.87	✓	0.93
Char.	`capitalize`	0.47	0.60	×	0.40	–	0.07

Layer localization matches steering optimality. Across tasks with high recovery, the layers where patching first crosses 0.5 recovery cluster with the optimal FV-injection layers: Llama Base L12–L18

(FV layer mean across tasks: L11.4); Llama Instruct L24–L32 (FV mean: L11.5—note: two distinct ranges); Gemma Base/IT L22–L36 (FV mean: L21.3); Mistral L14–L22 (FV mean: L13.2). This convergence between two independent measurements (where FV injection is most effective vs. where cross-template transfer is causally concentrated) strengthens the interpretation that FVs operate at specific processing bottlenecks rather than via a diffuse mechanism.

Layer ranges for first-decodable. A novel observation we surface from these data: for tasks where the logit lens *eventually* decodes the answer, the first-decodable layer (mean top-10 ≥ 0.10) is consistently 10–25 layers later than the optimal FV-injection layer. On Llama Base, `country_capital` optimal FV layer is L6.5 but no logit-lens layer crosses 0.10 (gap = ∞); `plural` optimal FV layer is L8.0, first-decodable layer is L24 (gap of 16 layers). On Llama IT, `country_capital` optimal FV layer is L5.2, first-decodable layer is L30 (gap of 24.8 layers). Across all (task, model) pairs where both metrics are defined, the median gap is 12 layers and the maximum is 26 layers. This temporal dissociation—FV injection at the early-to-middle layers, logit-lens-readable answer emerging only after 10+ further layers of processing—is a striking quantitative consequence of FVs encoding a process rather than an answer. To our knowledge, this temporal-gap statistic across 12 tasks $\times$ 6 models has not been reported before; it complements the cross-temporal observation of She et al. [30] during training.

P Additional Figures

Figure 9 reports the full per-task FV steering vs. logit-lens comparison across all 6 models (the body's Figure 3 shows 4 representative tasks); Figure 10 reports the cosine–transfer scatter for all cross-template pairs across all 6 models; Figure 11 reports FV vocabulary projection coherence; Figure 12 aggregates the tuned-lens validation evidence; Figure 13 reports the activation-patching recovery heatmap; Figure 14 compares zero-shot, 5-shot ICL, and FV-steered accuracy across all 6 models.

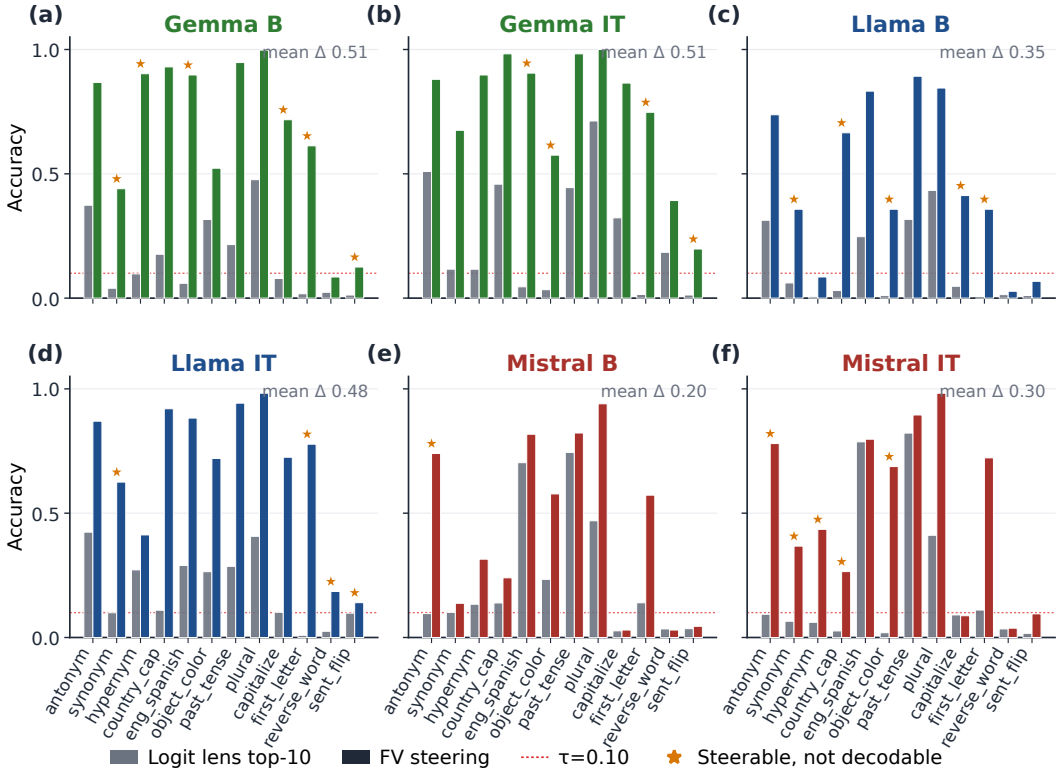

Figure 9: FV steering accuracy vs. logit-lens top-10 accuracy at the best layer for all 12 tasks across all 6 models. Gap annotations show the steerability–decodability difference; orange stars mark steerable-not-decodable tasks. For every task on every model, FV steering meets or exceeds logit-lens accuracy. The most dramatic gaps are `country_capital` (Llama Base: -0.82) and `first_letter` (Llama IT: -0.91). `sentiment_flip` readability uses polarity classification (App. Q).

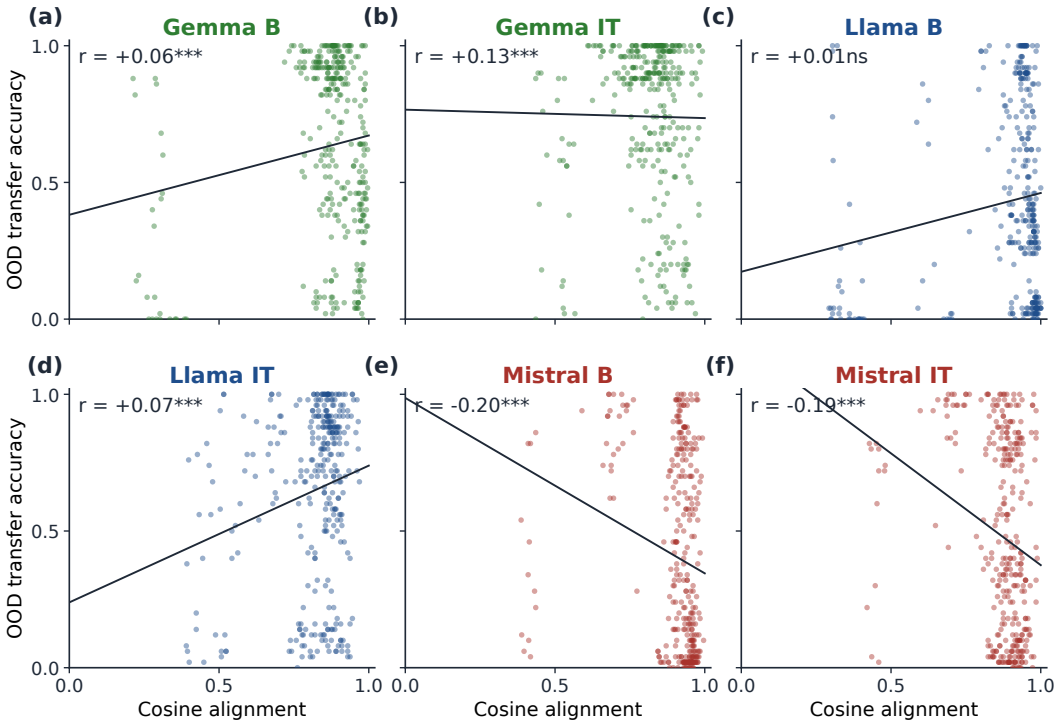

Figure 10: Cosine similarity vs. OOD transfer accuracy for all cross-template pairs across all 6 models, colored by task. Dashed lines mark the preliminary study's reported $r = -0.572$ scatter; current pooled correlations range from -0.199 (Mistral Base) to $+0.126$ (Gemma IT), dissolving the Simpson's paradox at 12-task scale. Points at high cosine (> 0.80) span the full range of transfer accuracy on every model, confirming that geometric alignment does not predict functional transfer.

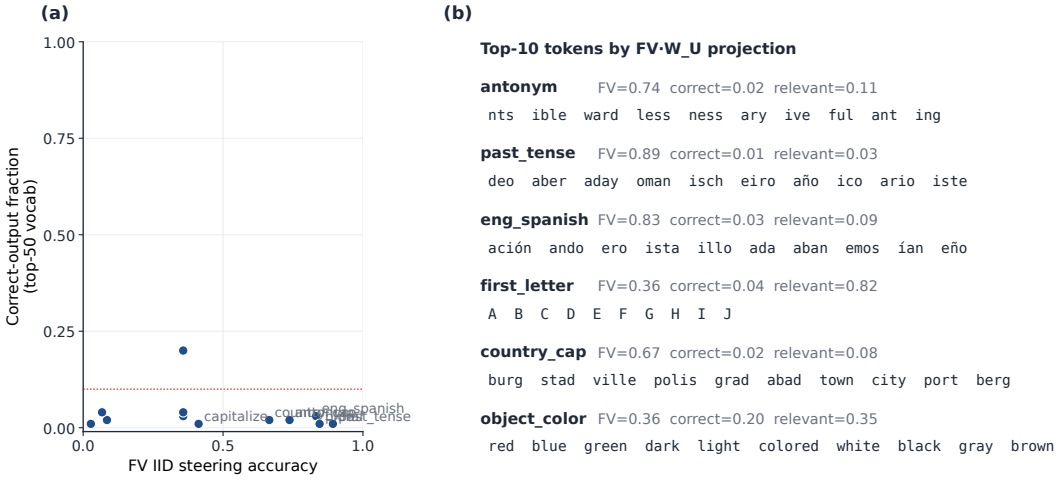

Figure 11: FV vocabulary projection analysis (Llama-3.1-8B Base). **(A)** Projection coherence vs. steering accuracy: if FVs were answer directions, points would lie on $y = x$ (dashed line); instead, FVs achieving >0.90 steering accuracy still project to incoherent tokens (bottom-right region). **(B)** Top-10 projected tokens for three representative tasks: `past_tense` projects to garbage (`deo`, `aber`, `aday`); `first_letter` is the partial exception, projecting to single-character tokens but not the correct specific letters; `antonym` projects to morphological fragments (`nts`, `ible`, `ward`). FVs encode computational instructions, not answer directions.

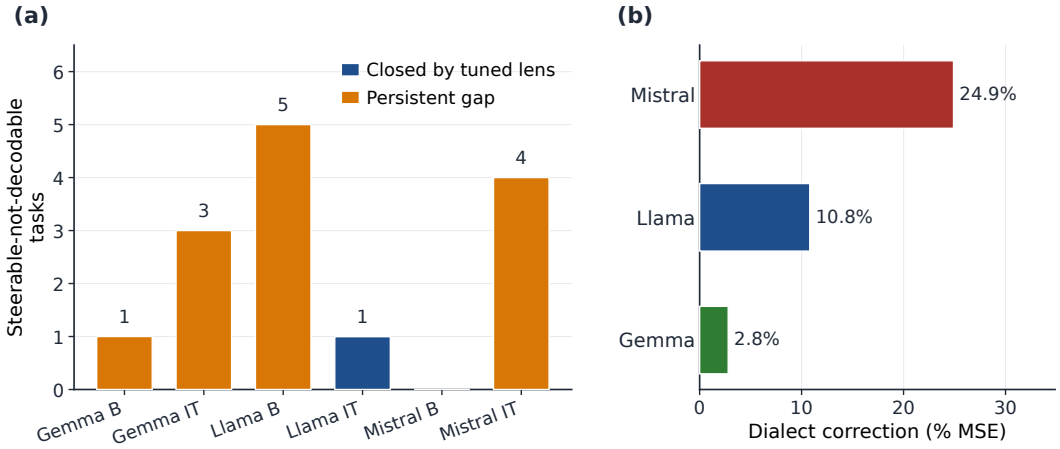

Figure 12: Tuned-lens validation. **(A)** Gap persistence: each square is one steerable-not-decodable task×model pair; red × indicates the gap persists under tuned-lens correction, green ✓ indicates the gap closed (only 1 of 14). **(B)** The expanded 2×3 matrix; the "tuned lens only" row contains a single borderline entry. **(C)** Dialect correction magnitude by model family (Gemma 2.8%, Llama 10.8%, Mistral 24.9%): despite a $\sim$10× range, all families show the same dissociation; correction anti-correlates with readability improvement ($r = -0.478$).

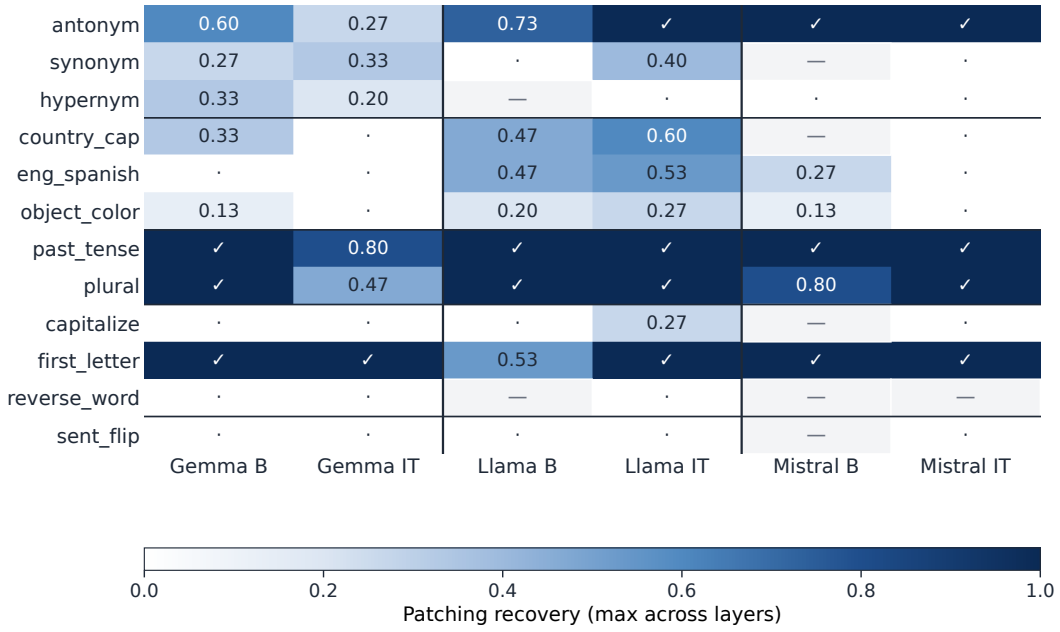

Figure 13: Activation-patching recovery accuracy (max across layers) for all tasks and models. ✓ = perfect recovery (≥ 0.99) at some layer; × = zero recovery (≤ 0.01) everywhere; light shading = task failed the IID gate. Easy tasks (`past_tense`, `plural`, `first_letter`) achieve perfect recovery on most models; hard tasks (`reverse_word`, `sentiment_flip`) show zero recovery everywhere—confirming that the difficulty hierarchy reflects causal localization, not statistical correlation.

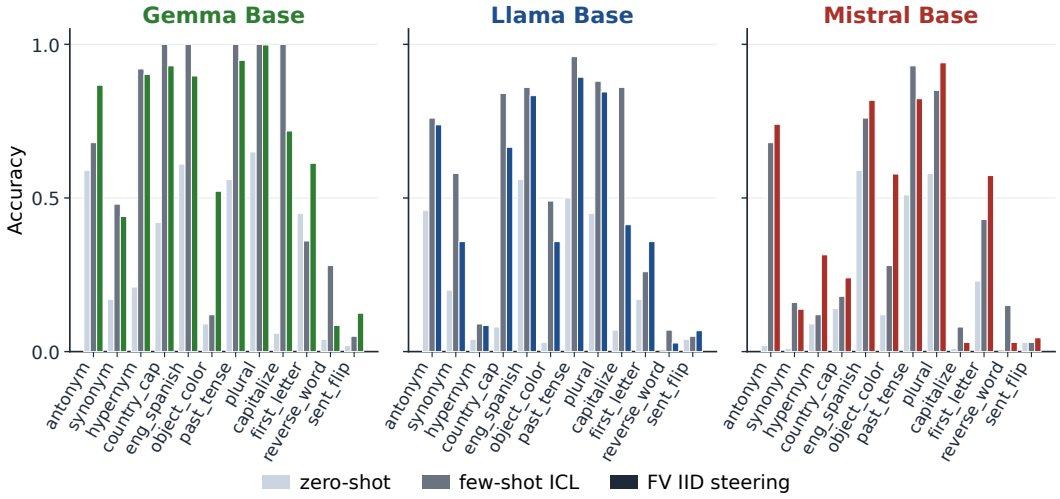

Figure 14: Zero-shot vs. 5-shot ICL vs. FV-steered accuracy across all 6 models. FV steering is consistently additive or neutral on every model: aggregate destructive cases (where FV steering reduces accuracy below zero-shot) are 0–1% across all 6 models. For most tasks, FV steering matches or approaches few-shot ICL accuracy. This aggregate-task statistic is compatible with prior reports of substantial per-sample anti-steerability [6, 33].

Q Statistical and Computational Details

Logit-lens computation. For each example i at layer ℓ, we compute $\hat{\mathbf{y}}_\ell^{(i)} = \mathrm{LayerNorm}_{\text{final}}(\mathbf{h}_\ell^{(i)}) \cdot W_U + \mathbf{b}_U$ and sort logits in descending order. A prediction counts as top-k correct if the correct output token's ID appears among the k highest-scoring token IDs. For multi-label tasks (`synonym`, `hypernym`, `object_color`), we accept any valid alternative's first token. The final LayerNorm and unembedding matrix are extracted directly from each model's parameters; no learned parameters are introduced.

Sentiment polarity (sentiment_flip). First-token accuracy is uninformative for `sentiment_flip` (both polarities can begin with arbitrary tokens). We construct a sentiment contrast vector $\mathbf{d}_{\text{sent}} = (\bar{\mathbf{e}}_{\text{neg}} - \bar{\mathbf{e}}_{\text{pos}})/\|\bar{\mathbf{e}}_{\text{neg}} - \bar{\mathbf{e}}_{\text{pos}}\|$, where $\bar{\mathbf{e}}_{\text{pos}}$ and $\bar{\mathbf{e}}_{\text{neg}}$ are mean token embeddings of the positive and negative first tokens in the sentiment-flip pairs. At each layer, we compute $\cos(\mathrm{LayerNorm}(\mathbf{h}_\ell), \mathbf{d}_{\text{sent}})$ for each example; polarity classification accuracy is the fraction with sign agreement to the expected polarity. We mark this metric explicitly in every figure and table caption that includes `sentiment_flip` readability.

Multiple comparisons. We apply Bonferroni correction wherever multiple hypothesis tests are conducted: within-task correlations ($k = 12$ tests, $\alpha_{\text{corrected}} = 0.05/12 = 0.0042$); logit-lens vs. FV paired tests ($k = 12$, $\alpha_{\text{corrected}} = 0.0042$); permutation tests for excess dissociation ($k = 12$ per model, same correction).

Permutation tests. For each task on each model we test whether the observed dissociation rate exceeds chance using 1,000 permutations of the alignment–transfer pairing within the task. The p-value is the fraction of permutations yielding a dissociation rate at least as large as observed; per-task p-values are Bonferroni-corrected as above.

Tokenizer handling. All three families (Llama BPE, Gemma SentencePiece, Mistral BPE) treat the leading-space variant of an output token differently. We encode every expected output token both with and without a leading space and use whichever produces a valid first token under each model's tokenizer. Worked examples of the encoding for each family: for `country_capital` "Paris," the Llama BPE tokenizer typically uses "␣Paris," Gemma SentencePiece uses "␣Paris," and Mistral BPE uses "␣Paris"—all three resolving to a leading-space variant for words that follow non-trivial prefix text. For `first_letter` "A," we use the bare "A" token under all three.

Generation defaults. Greedy decoding (no sampling), maximum 5 new tokens (10 for `sentiment_flip`, 3 for `first_letter`), end-of-sequence stopping enabled.

Precision and batching. Model weights are bfloat16; FV mean accumulators use float32 to minimize numerical error. Batch sizes follow GPU profile (128 on H100, 256 on H200). To amortize the α sweep, the input batch is tiled across α values in a single forward pass (batch of B prompts becomes a tiled batch of $B \cdot |\alpha\text{-strengths}|$ with per-strength multipliers applied at the residual-stream hook), producing all α-strengths' generations in one pass. This is numerically identical to per-strength calls but 3–8$\times$ faster.

R Compute Details

We additionally ran on NVIDIA H200 141GB for the largest steering sweeps; on H200 the steering-evaluation stage drops to $\sim$6–10 h per model, total $\sim$10–15 h. The full project (including preliminary experiments not reported in the main paper) consumed approximately 200 GPU-hours on H100 and 50 GPU-hours on H200.

Table 30: Compute budget per model on NVIDIA H100 80GB. Activation patching is the dominant cost; the total per model is 15–24 hours.

Stage	Estimated time	Hardware
Zero-shot + few-shot baselines	$\sim$1 h	GPU
FV extraction (1,536 FVs)	$\sim$3–4 h	GPU
Activation caching (zero-shot)	$\sim$0.5 h	GPU
Logit lens + FV vocab projection	$\sim$0.3 h	GPU
Steering evaluation ($\sim$98K configurations)	$\sim$10–18 h	GPU
Tuned-lens training + evaluation	$\sim$0.1 h	GPU
Total per model	**$\sim$15–24 h**	NVIDIA H100 80GB
Total (6 models)	**$\sim$4–6 days**	

S Worked Example: country_capital on Llama-3.1-8B Base

To make the methodology concrete, we walk through the full pipeline for a single (task, model) pair: `country_capital` on Llama-3.1-8B Base.

Baseline accuracy. Zero-shot accuracy: 0.08 (Llama Base only weakly retrieves capitals from bare prompts). Five-shot ICL: 0.84.

FV extraction. For each of the 8 templates we extract per-layer FVs at all 32 residual-stream blocks. The IID-best layer averages L6.5 across templates, with templates T1 ("The capital of {X} is"), T4, T7, and T8 yielding strong FVs (per-template IID 0.84–0.88) and T2 ("{X}'s capital city is") yielding the weakest (IID 0.22).

IID steering accuracy. Mean across 8 templates at the best (α, layer) per template: 0.665 (Table 8). The FV redirects Llama Base's existing capability (zero-shot 0.08) to ICL-comparable accuracy.

Logit-lens decodability. For zero-shot prompts of `country_capital` on Llama Base, logit-lens top-10 accuracy is $<$0.01 at layers 1–20 and rises only weakly at late layers, peaking at L32 with mean top-10 = 0.056. The first-decodable layer (mean top-10 ≥ 0.10) does not exist for this task on this model: the logit lens never crosses the threshold.

Steerability–decodability gap. FV steering accuracy 0.880 (best template T4, L4) minus logit-lens best-layer top-10 0.056 = -0.824. This is the most dramatic steerability-without-decodability case in the entire 72-pair matrix.

Post-steering logit lens. After applying the FV at L4, the maximum post-steering logit-lens top-10 across all subsequent layers and templates is 0.078—near zero. The FV steers the output without leaving any logit-lens-visible trace at intermediate layers, consistent with a non-representational (modulatory) mechanism on Llama Base.

Tuned-lens correction. Per-layer diagonal translators are trained (mean improvement 11.4%, max 16.5%, r(depth, improvement) = $+0.898$). After applying the tuned lens, top-10 accuracy at the best layer is 0.056—unchanged. The dialect correction does not reveal hidden decodability.

Activation patching. Country_capital passes the IID gate, so patching is run. We sample 5 (source, target) template pairs, evaluate 50 IID examples per pair, and patch the residual stream at every layer. Recovery accuracy peaks at L12–L18 (max recovery 0.87 on T4$\rightarrow$T1), with mean across (case, layer) of 0.722. This places `country_capital` in the "mixed" patching category (Table 29): cross-template transfer is causally concentrated at specific layers in the recoverable range, but recovery is not perfect on every (source, target) pair.

Summary of the worked example. Country_capital on Llama Base is the cleanest illustration of our central claim: an FV that achieves 0.880 IID steering accuracy when applied at L4, redirects model behavior at specific layers (recoverable by patching), but is invisible to the model's own

unembedding at every intermediate layer (logit lens 0.056 at peak), survives a dialect-correction robustness check (tuned lens 0.056), and produces near-zero post-steering deltas (0.078) on the logit-lens reading. The FV encodes a computational instruction that the model executes through pathways the unembedding cannot decode.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract's three concrete claims, namely the universal steerability–decodability gap (Figure 2), 93% gap persistence under tuned-lens correction (Table 2), and dissolution of the cosine–transfer correlation ($\Delta R^2 \leq 0.011$, Table 1), map directly to results in §4 and §5. Contributions enumerated in §1 match the body sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: §6 ("Limitations" paragraph) discusses model size (7–9B parameters), single FV extraction method (mean-of-differences), short-output evaluation, the diagonal tuned-lens constraint (cannot capture off-diagonal rotations), and the absence of adversarial template phrasings. Each limitation is paired with a concrete next experiment that could test it.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper makes no formal-theorem claims; all results are empirical. The conceptual claim that the linear representation hypothesis decomposes into linear decodability and linear steerability (§6) is presented as an empirical decomposition supported by data, not as a formal theorem.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: §3 specifies the FV extraction and steering equations (Eq. 1), logit-lens and FV vocabulary projection through final LayerNorm and W_U, the diagonal tuned-lens translator form, and the IID-gating threshold. The IID-gating definition is restated formally in App. A (Definition 1). Appendix A lists all 96 templates verbatim, App. L specifies the tuned-lens training procedure (AdamW, lr 10^{-3}, weight decay 0.01, patience 15, max 200 epochs, 80/20 split, identity initialization), App. O specifies the patching protocol, and App. Q specifies tokenizer handling, generation defaults, multiple-comparisons correction, and permutation-test details.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code and a curated release of the task data will be made publicly available under an open license upon acceptance, anonymized to preserve double-blind review. The methodology specifications (App. A–Q) are intended to be sufficient to re-implement the pipeline from scratch; all 96 templates are listed verbatim, all hyperparameters and thresholds are stated, and the worked example in App. S traces a single (task, model) pair end-to-end.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: §3 specifies the α sweep ($\{0.5, 1.0, \ldots, 5.0\}$ with adaptive refinement), the layer sweep (all residual-stream blocks), the IID gate ($\tau = 0.10$), the contrast classes (15 positive ICL prompts vs. 15 zero-shot prompts), and the cross-template evaluation (56 directed pairs per task, 50 IID test prompts, 50 OOD test prompts per template). Appendix L specifies tuned-lens training hyperparameters; App. O specifies the patching case-selection strategy; App. Q specifies tokenizer handling and generation defaults.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report Bonferroni-corrected within-task Pearson correlations (App. F, $\alpha_{\text{corrected}} = 0.0042$ for the 12-task family); paired t-tests for logit-lens vs. tuned-lens readability (App. L, Table 19); permutation tests for excess dissociation (1,000 shuffles per task per model, App. G); and Pearson correlations with p-values for post-steering delta vs. IID accuracy (App. N, Table 28). The hierarchical regression (Table 1) reports R^2 and ΔR^2 across $n \in [10{,}752, 14{,}112]$ cross-template pairs per model. Statistical procedures are specified in App. Q.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Appendix R (Table 30) reports per-stage compute on NVIDIA H100 80GB: total per model 15–24 hours, total for 6 models 4–6 days. The dominant cost is steering evaluation ($\sim$98K configurations, 10–18 hours per model). H200 141GB reduces total per-model time to 10–15 hours. Total project compute including preliminary experiments: $\sim$200 H100-hours and 50 H200-hours.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research uses publicly released language models under their respective licenses, evaluates them on benign linguistic and factual tasks, involves no human subjects or crowdsourced annotation, releases no new generative models or scraped datasets, and is conducted in accordance with the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: §6 ("Implications for AI safety") discusses the negative impact: standard interpretability tools (logit lens, vocabulary projection) may fail to detect FV-style steering interventions, especially on Llama and Gemma models. Practitioners relying on these tools for safety monitoring may miss steering attacks. Positive impacts include a clearer evaluation protocol for steering interventions and methodological tools (the 2×2 matrix, IID gating, tuned-lens robustness check) that other researchers can adopt to validate their own steering vectors before deployment.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: We release no new pre-trained models, no scraped datasets, and no high-misuse-risk artifacts. The task data are small curated lists of single-token input–output pairs (e.g., word–antonym pairs) that pose no foreseeable misuse risk.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We use Llama-3.1-8B (Base and Instruct) under Meta's Llama 3.1 Community License, Gemma-2-9B (Base and IT) under Google's Gemma Terms of Use, and Mistral-7B-v0.3 (Base and Instruct) under the Apache 2.0 license. All models are publicly released; we cite the model providers and use them only for research evaluation, in compliance with their respective terms.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We document the 12 task data files (input–output pairs) and 96 templates that constitute the new evaluation battery. All 96 templates are listed verbatim in App. A; pre-registered difficulty predictions are documented in App. K; per-task pair counts and evaluation methods are in Table 3. The task data and templates will be released alongside code, with a structured README documenting their provenance and intended use.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper involves no crowdsourcing and no human subjects; all evaluation is automated against pre-curated input–output pairs.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper involves no human subjects, so no IRB review is applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are the *object* of study in this paper (we evaluate steering vectors on Llama-3.1-8B, Gemma-2-9B, and Mistral-7B-v0.3), not a methodological component used to produce or analyze results. No LLM is used as a research tool to generate or evaluate experimental data.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.