RAPTOR: Ridge-Adaptive Logistic Probes

Ziqi Gao, Yaotian Zhu, Qingcheng Zeng, Xu Zhao, Ziqing Wang, Feng Ruan, Kaize Ding
Northwestern University
ziqigao2026@u.northwestern.edu kaize.ding@northwestern.edu

Abstract

Probing studies what information is encoded in a frozen LLM's **layer representations** by training a lightweight predictor on top of it. Beyond analysis, probes are often used operationally in *probe–then–steer* pipelines: a learned **concept vector** is extracted from a probe and then injected via additive activation steering by adding it to a layer representation during the forward pass. The effectiveness of this pipeline hinges on estimating concept vectors that are accurate, directionally stable under ablation, and inexpensive to obtain. Motivated by these desiderata, we propose RAPTOR (**R**idge-**A**da**pt**ive L**o**gistic P**r**obe), a simple ℓ_2-regularized logistic probe whose validation-tuned ridge strength yields concept vectors from normalized weights. Across extensive experiments on instruction-tuned LLMs and human-written concept datasets, RAPTOR matches or exceeds strong baselines in accuracy while achieving competitive directional stability and substantially lower training cost; these quantitative results are supported by qualitative downstream steering demonstrations. Finally, using the Convex Gaussian Min–max Theorem (CGMT), we provide a mechanistic characterization of ridge logistic regression in an idealized Gaussian teacher–student model in the high-dimensional few-shot regime, explaining how penalty strength λ mediates probe accuracy and concept-vector stability, yielding structural predictions that qualitatively align with trends observed on real LLM embeddings.

1 Introduction

Probing trains lightweight predictors on frozen LLM layer representations to identify whether a target concept is decodable from those representations [1, 2, 3, 4]. Beyond diagnosis, probes are increasingly used operationally: the learned probe can guide downstream interventions.

One such downstream intervention is *steering*, defined as the modulation of behavior at inference time without updating model weights [5, 6, 7]. Within this domain, we focus on *additive activation steering*, a technique that modifies a layer's representation by injecting a learned direction [8, 9]. For an input sentence x tokenized as $(t_1, \ldots, t_T)$, let $h_{\ell,T} \in \mathbb{R}^p$ denote the **layer representation** of the last token at layer $\ell \in \{1, \ldots, L\}$. Consistent with standard probing practices, we treat $h_{\ell,T}$ as a sentence-level summary. Additive activation steering intervenes by directly editing this representation: $h_{\ell,T} \leftarrow h_{\ell,T} + \alpha\, v_\ell$, where $v_\ell \in \mathbb{R}^p$ represents the **concept vector** for the target concept at layer ℓ, and $\alpha \in \mathbb{R}$ controls the steering strength. This approach is particularly advantageous as it is simple to implement and imposes negligible inference overhead. However, its effectiveness is contingent upon the quality of v_ℓ (and α); if the estimated concept vector is noisy or brittle, the resulting steering becomes unreliable.

What is a good probe? For probe–then–steer applications, a good probe must satisfy three requirements: **(i) accuracy**, so that the concept is reliably decoded from layer representations; **(ii) directional stability**, so that the learned vector remains reusable under small data perturbations rather than reflecting idiosyncrasies of a particular split [10, 11]; and **(iii) computational efficiency**,

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

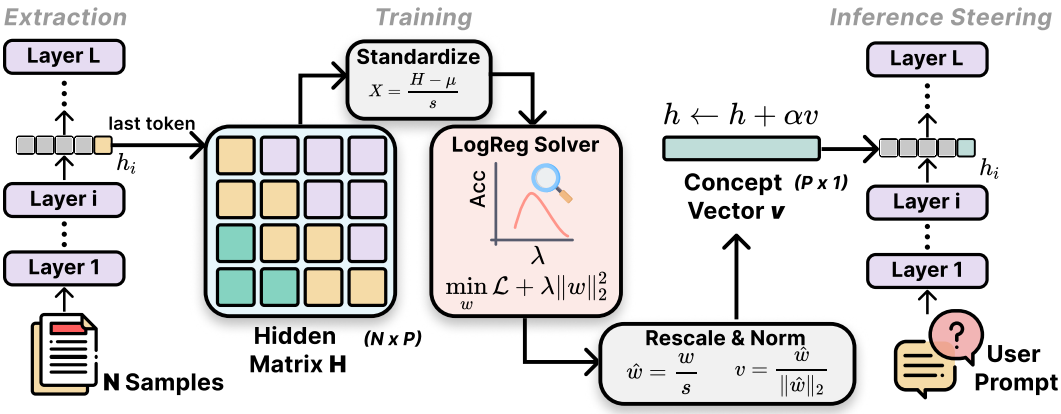


Figure 1: **Overview of the RAPTOR pipeline for additive activation steering:** extract layerwise last-token embeddings, standardize features, fit an ℓ_2-regularized logistic probe with λ selected on a validation split, rescale to embedding space and normalize to obtain the concept vector v_ℓ, then steer at inference by $h_{\ell,T} \leftarrow h_{\ell,T} + \alpha v_\ell$.

since probes are typically trained across many layers, concepts, and models. These criteria define our evaluation lens.

Motivated by these desiderata, we propose **RAPTOR**: **Ridge-Adaptive Logistic Probes**. For each tuple of model, layer, and concept, RAPTOR fits a validation-tuned ℓ_2-regularized logistic probe on frozen layer representations and uses its normalized weight vector as the concept vector v_ℓ. The single ridge parameter λ provides a simple mechanism for balancing predictive accuracy and directional stability, while the estimator remains computationally inexpensive. This minimalist design is especially useful for probe–then–steer pipelines, where higher probe accuracy alone does not necessarily yield a reliable intervention direction [12, 13, 14], and overly elaborate estimators can make exhaustive layer and model sweeps expensive [15].

LLM probing often lies in a high-dimensional few-shot regime, where the layer representation dimension p can be comparable to, or even exceed, the number of labeled examples n. In this regime, ℓ_2 regularization is not merely a numerical stabilizer: the ridge strength λ can directly shape both predictive accuracy and the stability of the learned direction. We therefore analyze ridge logistic regression in a high-dimensional Gaussian teacher–student model with $n, p \to \infty$ and $n/p \to \delta$. Using the Convex Gaussian Min–max Theorem (CGMT), we characterize how λ controls the signal-aligned and orthogonal components of the learned direction, yielding structural predictions that qualitatively match trends on real LLM embeddings.

We evaluate RAPTOR through a benchmark spanning instruction-tuned models, human-written concept datasets, and full network depth. Across accuracy, directional stability, and training cost, RAPTOR matches or exceeds representative alternatives while remaining simple to train. We further include downstream steering evaluations showing that the learned directions support reliable additive control.

Our main contributions are as follows:

- **Raptor algorithm.** We introduce a one-knob ridge-logistic probing pipeline with explicit hyperparameter selection of λ, producing interpretable concept vectors suitable for additive activation steering.
- **Operator-aligned evaluation criterion.** We formalize concept-vector estimation for additive activation steering and argue that probes should be evaluated under a joint objective capturing *accuracy*, *directional stability*, and *computational cost*. We instantiate this criterion through a multi-model, multi-dataset, multi-layer benchmark comparing RAPTOR against representative alternatives.
- **Theory-backed interpretation.** We provide a self-contained high-dimensional characterization of ridge logistic regression that explains how λ mediates accuracy and concept-vector stability in the proportional regime.

2 Methodology

2.1 Related work

Probing studies what information is encoded in a model's *layer representations* by training an auxiliary predictor on top of frozen activations [1, 2, 4, 3, 15]. In this work, probing is not only diagnostic but also operational: the learned probe is used to estimate a concept direction for additive activation steering. For a concept c, model M, and layer ℓ, we collect last-token layer representations $\{h_i^{(\ell)}\}_{i=1}^N$, where $h_i^{(\ell)} \in \mathbb{R}^p$, together with binary labels $y_i \in \{0, 1\}$. The goal is to estimate a layer-wise concept vector $v^{(\ell)} \in \mathbb{R}^p$ that can be injected into the native hidden-state space through an intervention of the form $h^{(\ell)} \leftarrow h^{(\ell)} + \alpha v^{(\ell)}$.

Existing probe estimators expose different tradeoffs for this purpose. Linear probes are simple, interpretable, and directly yield a single direction, but can be brittle in high-dimensional, limited-data regimes. Nonlinear probes can improve predictive accuracy, but they do not naturally produce a single reusable concept vector. Related methods interpret probe weights as concept directions, as in TCAV [16], or use learned directions for editing and removal, including iterative nullspace-based methods [17] and amnesic removal [18]. Contrastive Activation Addition (CAA) uses the classical mean-difference approach to estimate steering directions [9]. More recent concept-estimation methods represent concepts as subspaces or distributions over directions, such as Gaussian Concept Subspace (GCS) [19] and estimators based on Recursive Feature Machines (RFMs) [20, 19].

In probe–then–steer pipelines, however, the intervention form is already fixed: one adds a learned direction to an internal activation. Thus the central methodological question is how to estimate a direction that is accurate, stable under small training perturbations, and inexpensive to obtain. RAPTOR targets this regime with a one-knob ridge-logistic probe: it preserves the operational simplicity of a linear concept vector while using validation-tuned ℓ_2 regularization to improve stability and optimization behavior.

2.2 Ridge-Adaptive Logistic Probe

Our choice of ridge logistic regression is motivated by the high-dimensional, small-sample regime typical of LLM probing: labeled hidden states are often nearly linearly separable, as shown empirically in Table 4. In this setting, the unregularized logistic loss may fail to attain its infimum at finite parameter values. Under complete separation, scaling a separating parameter pair (w, b) sends its norm to infinity while the loss decreases to zero. This can lead to large-norm, poorly conditioned, and numerically unstable solutions. When both classes are present in the training data, adding an ℓ_2 penalty to the weights restores a finite solution and introduces a single interpretable regularization knob. We therefore select the ridge strength λ by validation for each model–layer–concept tuple.

Method setup. Fix a concept c, model M, and layer ℓ. We form a labeled dataset $\{(h_i^{(\ell)}, y_i)\}_{i=1}^N$ with $h_i^{(\ell)} \in \mathbb{R}^p$ and $y_i \in \{0, 1\}$, and use a fixed stratified split into training, validation, and test index sets $\mathcal{I}_{\text{tr}}$, $\mathcal{I}_{\text{val}}$, and $\mathcal{I}_{\text{te}}$. Let $\tilde{y}_i = 2y_i - 1 \in \{\pm 1\}$.

For each layer, we standardize embeddings using training-set statistics only:

$$\mu_j^{(\ell)} = \frac{1}{|\mathcal{I}_{\text{tr}}|} \sum_{i \in \mathcal{I}_{\text{tr}}} h_{ij}^{(\ell)}, \qquad s_j^{(\ell)} = \sqrt{\frac{1}{|\mathcal{I}_{\text{tr}}|} \sum_{i \in \mathcal{I}_{\text{tr}}} \left(h_{ij}^{(\ell)} - \mu_j^{(\ell)}\right)^2}, \quad j = 1, \ldots, p. \tag{1}$$

Zero entries of $s^{(\ell)}$ are replaced by 1 in implementation. We then form standardized features

$$x_{ij}^{(\ell)} = \frac{h_{ij}^{(\ell)} - \mu_j^{(\ell)}}{s_j^{(\ell)}}, \qquad j = 1, \ldots, p, \tag{2}$$

and apply the same transform to validation and test embeddings. This train-only standardization avoids information leakage and ensures that ridge tuning is not dominated by a small number of high-variance coordinates.

Given standardized features, RAPTOR fits ridge-regularized logistic regression with an intercept. For an index set $\mathcal{I} \subseteq \{1, \dots, N\}$ and ridge strength $\lambda > 0$, define

$$\mathcal{L}_\lambda^{(\ell)}(w, b; \mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \log\Big(1 + \exp\big(-\tilde{y}_i(w^\top x_i^{(\ell)} + b)\big)\Big) + \frac{\lambda}{2}\|w\|_2^2, \tag{3}$$

where $(w, b) \in \mathbb{R}^p \times \mathbb{R}$.

We choose λ on the validation split:

$$\lambda^\star = \arg\max_{\lambda \in \Lambda} \mathrm{Acc}_{\mathrm{val}}(\lambda), \tag{4}$$

then refit the probe on $\mathcal{I}_{\mathrm{tr}} \cup \mathcal{I}_{\mathrm{val}}$ using the sample-size-adjusted ridge strength $\lambda^\star \frac{|\mathcal{I}_{\mathrm{tr}}|}{|\mathcal{I}_{\mathrm{tr}} \cup \mathcal{I}_{\mathrm{val}}|}$. Let the resulting standardized-space parameters be $(\hat{w}^{(\ell)}, \hat{b}^{(\ell)})$. To obtain a concept vector in the native embedding space, we fold the parameters back through the standardization map: $\hat{\omega}_j^{(\ell)} = \frac{\hat{w}_j^{(\ell)}}{s_j^{(\ell)}}$, $\hat{b}_{\mathrm{orig}}^{(\ell)} = \hat{b}^{(\ell)} - \langle \hat{\omega}^{(\ell)}, \mu^{(\ell)} \rangle$. The original-space probe logit is therefore $(\hat{\omega}^{(\ell)})^\top h^{(\ell)} + \hat{b}_{\mathrm{orig}}^{(\ell)}$. For additive activation steering, we use the normalized concept direction $v^{(\ell)} = \frac{\hat{\omega}^{(\ell)}}{\|\hat{\omega}^{(\ell)}\|_2}$.

2.3 Steering Method

Given a learned layer-wise concept direction, additive steering only requires choosing the injection strength. We adopt the *GCAV* per-sample calibration rule [21, 22] as an off-the-shelf way to set this strength without retraining. Fix a layer ℓ and let $h^{(\ell)} \in \mathbb{R}^p$ denote the (pre-steering) hidden activation. We steer by $h_{\mathrm{st}}^{(\ell)} = h^{(\ell)} + \alpha\, v^{(\ell)}$. Along the direction $v^{(\ell)}$, the detector logit changes affinely:

$$(\omega^{(\ell)})^\top h_{\mathrm{st}}^{(\ell)} + \beta^{(\ell)} = \underbrace{(\omega^{(\ell)})^\top h^{(\ell)} + \beta^{(\ell)}}_{=:\, g^{(\ell)}(h)} + \alpha\, \|\omega^{(\ell)}\|_2. \tag{5}$$

Given a target probability level $p_0 \in (0, 1)$ and $g_0 \coloneqq \mathrm{logit}(p_0)$, GCAV sets the *minimal* strength that satisfies the target:

$$\alpha_{\mathrm{amplify}}^{(\ell)}(h) = \max\left\{0,\ \frac{g_0 - g^{(\ell)}(h)}{\|\omega^{(\ell)}\|_2}\right\}, \quad \alpha_{\mathrm{remove}}^{(\ell)}(h) = \min\left\{0,\ \frac{g_0 - g^{(\ell)}(h)}{\|\omega^{(\ell)}\|_2}\right\} \tag{6}$$

In words, we inject only when the current activation does not already meet the target; the required strength is computed in closed form from the probe logit.

3 Experiments

We evaluate probe quality for probe–then–steer pipelines along three axes: (i) *classification accuracy* on real LLM embeddings, (ii) *directional stability* of the learned concept axis under small training perturbations, and (iii) *computational cost*. All methods operate on the same layerwise embeddings and share the same data split within each seed.

3.1 Setup

We benchmark RAPTOR against xRFM, GCS, logistic regression without regularization tuning (Fixed-reg), and a raw mean-difference baseline with a Euclidean midpoint readout (CAA) on six instruction-tuned LLMs from the Llama, Qwen, and Gemma families, and six human-written binary concept datasets: STSA [23], Cities [24], Common [24], CounterFact [25], HateXplain [26], and Sarcasm [27]. For each model–dataset pair, we extract all-layer last-token embeddings and train probes independently per layer. We use five stratified splits (seeds 42–46), each with approximately 20% test, 16% validation, and 64% training data. Because concept predictability varies by layer, we report both average test accuracy over layers and test accuracy at the layer with the highest validation accuracy, averaged over the five seeds; sample standard deviations are reported in the appendix. RAPTOR tunes λ on validation data and reports refit test performance; Fixed-reg follows the same procedure without tuning, using $\lambda = 1/|\mathcal{I}_{\mathrm{tr}}|$ before refitting. CAA re-estimates class means on $\mathcal{I}_{\mathrm{tr}} \cup \mathcal{I}_{\mathrm{val}}$; xRFM and GCS follow their original pipelines.

Table 1: **Probe accuracy (%) across models and datasets.** Results are presented as **Average over layers** (Best Layer). All values are means over five seeds; best layers are selected using validation accuracy. The **purple background** highlights the highest average-over-layers accuracy within each model and column, including all ties at the displayed precision. The vertical line separates individual task results from the overall average across all six tasks.

Model	Method	STSA	Cities	Common	Counterfact	HateXplain	Sarcasm	Average
				Qwen 2.5 Series				
Qwen-3B-Instruct	GCS	83.5 (92.6)	84.1 (99.4)	62.5 (67.9)	64.7 (77.6)	66.0 (69.8)	81.1 (87.9)	73.6 (82.5)
	xRFM	87.8 (93.4)	**91.6** (99.5)	**69.0** (72.0)	66.6 (75.8)	70.3 (74.3)	85.2 (91.0)	**78.4** (84.4)
	Fixed-reg	84.6 (91.7)	87.4 (99.6)	61.7 (65.2)	63.1 (73.4)	68.7 (71.3)	81.5 (88.9)	74.5 (81.7)
	CAA	78.1 (91.4)	84.3 (99.7)	64.1 (68.6)	62.9 (77.8)	62.6 (68.5)	74.2 (82.7)	71.0 (81.4)
	Raptor (Ours)	**88.1** (93.7)	88.1 (99.5)	68.6 (71.6)	**68.8** (80.0)	**70.9** (74.1)	**85.3** (90.9)	78.3 (85.0)
Qwen-7B-Instruct	GCS	87.3 (93.9)	87.5 (99.5)	66.0 (70.8)	68.6 (80.0)	69.2 (72.3)	84.1 (89.4)	77.1 (84.3)
	xRFM	89.8 (94.2)	**93.7** (99.3)	**69.6** (73.0)	69.3 (80.3)	72.4 (75.2)	**87.5** (92.3)	**80.4** (85.7)
	Fixed-reg	87.8 (93.2)	89.7 (99.3)	63.3 (65.6)	65.2 (75.2)	67.7 (69.1)	85.3 (90.9)	76.5 (82.2)
	CAA	80.7 (92.0)	86.6 (98.8)	64.7 (69.7)	65.6 (78.3)	64.2 (70.0)	78.3 (85.3)	73.4 (82.3)
	Raptor (Ours)	**90.1** (94.6)	90.9 (99.4)	**69.6** (73.4)	**71.3** (81.5)	**73.0** (75.2)	**87.5** (92.2)	**80.4** (86.0)
Qwen-32B-Instruct	GCS	88.0 (94.1)	91.5 (99.5)	68.0 (71.9)	73.3 (82.8)	70.8 (73.6)	86.2 (91.5)	79.6 (85.6)
	xRFM	90.4 (94.9)	**95.5** (98.9)	**71.4** (74.4)	74.8 (82.2)	73.4 (75.9)	**89.4** (93.8)	**82.5** (86.7)
	Fixed-reg	89.1 (94.0)	93.4 (99.2)	65.4 (68.0)	71.7 (81.8)	66.3 (68.3)	88.0 (92.8)	79.0 (84.0)
	CAA	80.9 (89.4)	90.1 (98.9)	66.3 (70.3)	67.7 (80.4)	64.9 (70.2)	81.1 (87.9)	75.2 (82.8)
	Raptor (Ours)	**90.7** (94.8)	93.5 (98.8)	71.1 (74.5)	**76.2** (83.9)	**74.0** (75.9)	89.3 (93.3)	**82.5** (86.9)
				Gemma Series				
Gemma-7B-it	GCS	85.6 (91.9)	87.4 (98.6)	65.3 (70.2)	68.1 (78.3)	67.7 (71.0)	81.3 (87.4)	75.9 (82.9)
	xRFM	89.0 (93.3)	**93.1** (98.9)	**69.7** (72.8)	69.3 (77.5)	71.4 (75.2)	**85.1** (88.7)	**79.6** (84.4)
	Fixed-reg	87.2 (91.9)	88.9 (98.8)	63.2 (65.8)	65.3 (74.7)	68.1 (70.2)	82.6 (88.3)	75.9 (81.6)
	CAA	83.9 (90.5)	87.2 (98.4)	65.8 (69.6)	66.1 (77.6)	62.5 (67.4)	78.2 (84.3)	73.9 (81.3)
	Raptor (Ours)	**89.2** (93.2)	90.6 (98.8)	69.5 (72.6)	**71.2** (80.7)	**71.8** (75.4)	**85.1** (89.6)	**79.6** (85.0)
				Llama Series				
Llama-3.1-8B-Instruct	GCS	89.7 (93.7)	92.7 (99.8)	67.8 (71.5)	75.1 (87.3)	71.7 (73.6)	87.0 (91.0)	80.7 (86.1)
	xRFM	91.7 (94.6)	**96.8** (99.5)	**71.8** (74.3)	78.9 (87.3)	74.2 (76.7)	**90.1** (93.5)	**83.9** (87.7)
	Fixed-reg	90.0 (92.9)	93.7 (99.8)	65.5 (67.4)	76.1 (85.5)	67.6 (69.8)	88.8 (93.2)	80.3 (84.8)
	CAA	83.8 (92.5)	88.4 (97.9)	66.9 (70.2)	68.2 (81.2)	66.6 (70.7)	82.7 (87.2)	76.1 (83.3)
	Raptor (Ours)	**91.9** (94.2)	94.4 (99.5)	71.4 (73.5)	**80.0** (88.6)	**75.0** (77.0)	**90.1** (93.4)	83.8 (87.7)
Llama-3.1-70B-Instruct	GCS	91.1 (93.9)	95.1 (99.7)	70.9 (73.2)	80.6 (88.9)	75.0 (76.7)	91.3 (94.9)	84.0 (87.9)
	xRFM	93.0 (94.8)	**97.7** (99.5)	**73.7** (76.0)	84.0 (90.6)	76.8 (78.8)	93.7 (96.5)	**86.5** (89.4)
	Fixed-reg	92.0 (93.6)	96.1 (99.9)	68.7 (70.4)	81.9 (89.6)	71.0 (72.6)	93.3 (96.0)	83.9 (87.0)
	CAA	82.3 (90.1)	93.1 (99.3)	68.1 (70.7)	71.2 (83.4)	68.7 (72.9)	88.0 (92.9)	78.6 (84.9)
	Raptor (Ours)	**93.3** (94.7)	96.5 (99.9)	73.6 (74.6)	**84.6** (90.9)	**77.4** (79.0)	**93.8** (96.5)	**86.5** (89.3)

3.2 Probe accuracy

Table 1 reports probe accuracy over the full 6×6 model–dataset grid, with each cell showing layer-average test accuracy and validation-selected best-layer test accuracy in parentheses. RAPTOR improves over GCS in all 36 settings for layer-average accuracy and in $33/36$ settings for validation-selected best-layer accuracy. Compared with xRFM, RAPTOR matches or exceeds validation-selected best-layer accuracy in $17/36$ settings (16 wins and 1 tie), and improves layer-average accuracy in $21/36$ settings. Averaged over the grid, RAPTOR reaches 86.64% validation-selected best-layer accuracy, compared with 84.89% for GCS (+1.75 percentage points) and 86.37% for xRFM (+0.28 points). The corresponding layer-average accuracies for RAPTOR and xRFM are 81.85% and 81.88%, respectively. Compared with Fixed-reg and CAA, the mean gains are 3.52 and 7.15 percentage points for layer-average accuracy and 3.08 and 3.96 points for validation-selected best-layer accuracy, respectively. On challenging semantic datasets such as HateXplain and Sarcasm, RAPTOR consistently achieves higher mean best-layer accuracy than GCS, Fixed-reg, and CAA across all six models. Figure 2 visualizes per-setting differences (RAPTOR$-$baseline), where purple indicates improvements and green indicates decreases. For completeness, Algorithm 1 provides detailed pseudocode and the full training/validation protocol; code and experiment scripts will be released upon publication.

3.3 Directional robustness

We evaluate whether each method produces a stable concept axis under small perturbations of the concept-labeled training data. We evaluate 18 model–dataset settings spanning Qwen2.5-3B-Instruct, Qwen2.5-32B-Instruct, and Gemma-7B-it across all six datasets, covering 768 layers. For each (model, dataset, layer, method), we perform $K = 20$ ablation runs. In each run, we randomly drop 20% of the samples in $\mathcal{I}_{\mathrm{tr}} \cup \mathcal{I}_{\mathrm{val}}$, re-split the remainder into training and validation sets with

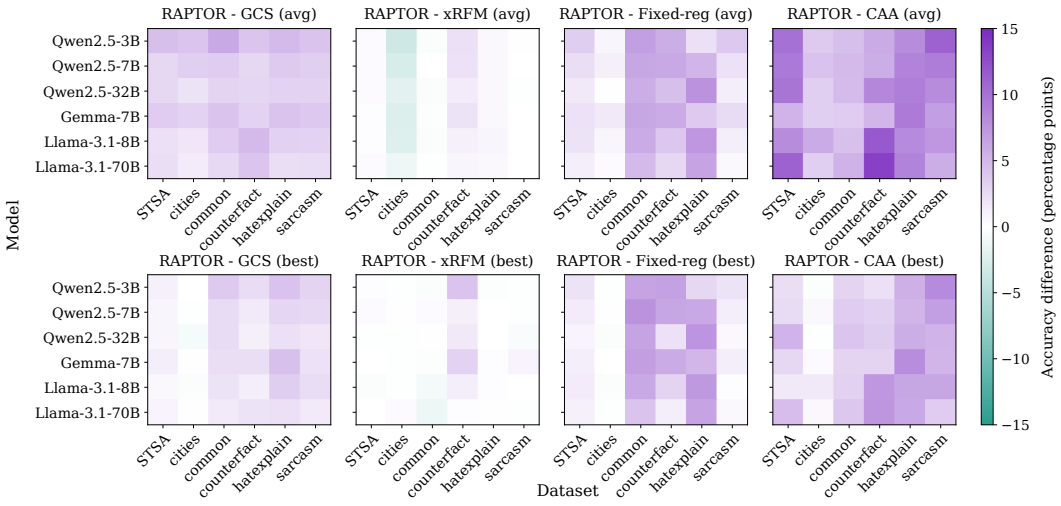


Figure 2: **Accuracy differences across the full model–dataset grid** (Raptor minus baseline)

validation fraction 0.2, and re-estimate the concept directions. RAPTOR re-selects λ on each ablated validation split, whereas Fixed-reg uses a predeclared regularization setting in every ablation. xRFM and GCS follow their original pipelines, while CAA re-estimates the raw mean-difference direction on the retained pool.

Let $\{v_r\}_{r=1}^{K}$ be the unit-normalized directions, excluding intercepts, returned across ablations for a fixed (model, dataset, layer). We measure axis stability irrespective of sign using mean absolute pairwise cosine similarity: $\text{Robust} = \frac{2}{K(K-1)} \sum_{1 \le r < s \le K} |\langle v_r, v_s \rangle|$. We report the mean and maximum of this score across layers in Figure 3, with diamonds indicating medians across the 18 settings. xRFM retains standardized-space directions, whereas the other methods use raw embedding coordinates. The top and bottom panels show maximum and mean robustness across layers, respectively. Each circle represents one model–dataset setting.

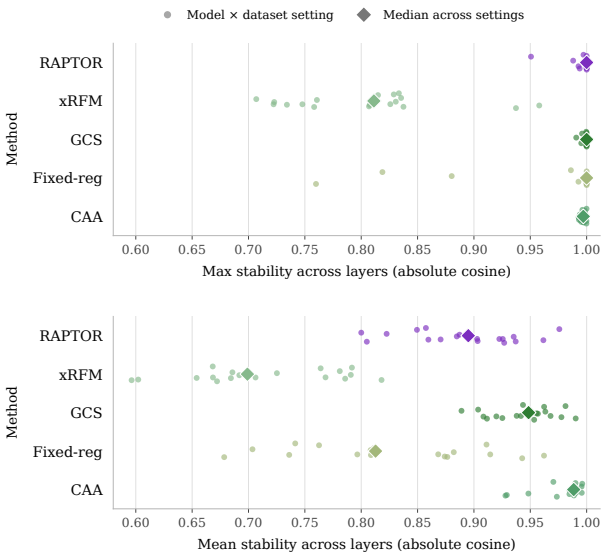


Figure 3: Directional stability.

Across the evaluated settings, RAPTOR consistently improves over xRFM. GCS and CAA remain more stable on average across layers, consistent with their use of averaging, but RAPTOR is close in maximum-layer robustness (the median maxima across the 18 settings round to 1.00 for all three methods). Compared with Fixed-reg, RAPTOR also has higher median layer-average robustness (0.90 versus 0.81). At the individual-setting level, RAPTOR achieves higher layer-average robustness than Fixed-reg in 15 of the 18 settings. This advantage holds for a majority of datasets within each of the three evaluated models. Thus, RAPTOR yields concept axes that are less sensitive to finite-sample perturbations than xRFM while retaining high predictive accuracy, a prerequisite for directional stability to be practically useful.

3.4 Computational cost

We measure training cost as wall-clock time under the same hardware and data pipeline. Each model–dataset–layer probe fit used one CPU worker. Since probes are trained independently per

layer, total fitting cost increases with the number of evaluated layers. Across the full grid, RAPTOR is consistently faster than xRFM and GCS. Figure 4 summarizes per-layer training time for all five methods across the same 36 model–dataset settings.

For each (dataset, model) pair, we plot the per-layer time summary for each method, with diamonds indicating medians across settings. CAA is particularly fast because its direction and midpoint readout are computed directly from class-conditional embedding means, without fitting a regression model. Fixed-reg is also fast because it fits logistic regression with a preset regularization setting, avoiding the penalty-parameter search used by RAPTOR. Although RAPTOR is slower than CAA and Fixed-reg, their lower accuracy on semantically demanding datasets (subsection 3.2) highlights the importance of predictive accuracy as the primary criterion for evaluating probes. RAPTOR nevertheless remains substantially faster than GCS and xRFM across the evaluated settings.

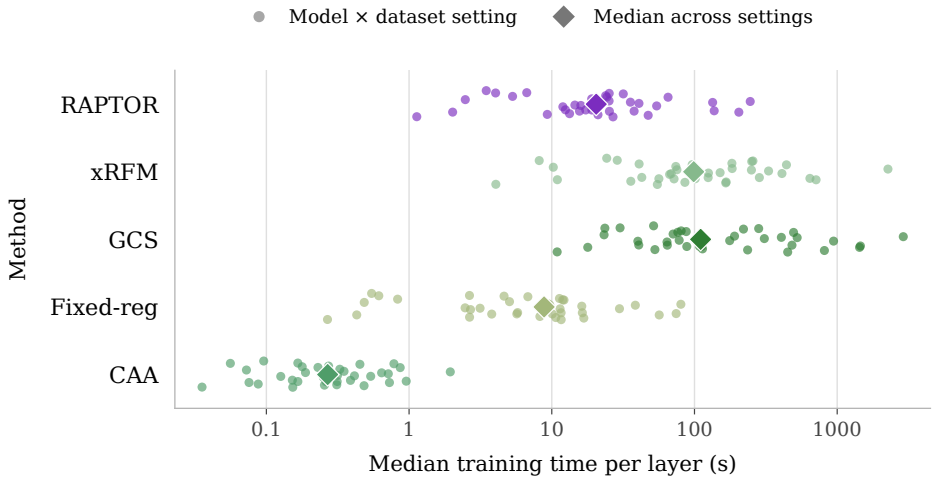


Figure 4: **Per-layer probe training time across 6×6 model–dataset settings.** The horizontal axis uses a logarithmic scale. Each circle shows the per-layer training-time summary for one model–dataset setting. Diamonds indicate medians across the 36 settings for each method.

3.5 Steering results

We evaluate steering control using the concept vectors learned by our probes. For a given concept and a target direction (either towards the concept or away from it), we intervene on the model by adding a scaled concept vector to the layers' representation. We use an adaptive per-layer steering strength α_ℓ chosen to drive the probe probability p_m toward an extreme target (towards: $p_m \approx 0.9999$; away: $P_m \approx 0.0001$). To avoid intervening with poorly aligned directions when the probe is unreliable, we optionally skip layers whose probe validation accuracy falls below a reliability threshold τ(e.g., $\tau = 0.7$ for Counterfact and $\tau = 0.8$ for STSA-positive in our runs).

Table 2: **Steering controllability.**

Dataset	Dir.	Succ.	Intv.	$\lvert\alpha\rvert$ med	$\lvert\alpha\rvert$ p90
CounterFact	away	1.000	0.833	10.50	66.50
CounterFact	towards	1.000	0.556	6.88	59.25
HateXplain	towards	1.000	0.654	6.80	49.00
HateXplain	away	1.000	0.538	4.70	24.50
Sarcasm	towards	1.000	0.715	8.83	28.00
Sarcasm	away	1.000	0.631	6.20	30.75
STSA	away	1.000	0.615	3.57	21.25
STSA	towards	1.000	0.792	12.38	29.25

We report three control metrics: (i) **probe-target success rate**, the fraction of evaluated layer–prompt pairs whose final probe probability lands in the desired extreme region (towards: $P_m \geq 0.9999$; away: $P_m \leq 0.0001$); (ii) **intervention rate**, the fraction of evaluated layer–prompt pairs where steering is actually applied (i.e., the baseline is not already in the target region), and (iii) **steering strength**, summarized by the distribution of $|\alpha_\ell|$ (median/p90/max).

Table 2 shows adaptive steering achieves near-perfect probe-coordinate control across all datasets and directions, while the required intervention rate varies by task (roughly 0.54 to 0.83 of pairs).

The steering strength exhibits a pronounced long tail: although median $|\alpha|$ is modest (about 3.6 to 12.4), some settings require very large interventions (max $|\alpha|$ up to 249), with a markedly heavier tail for away-direction control. We further break down the 'hard' layers that dominate the tail in Appendix B.6, and provide detailed pseudocode for our steering-strength selection procedure in Algorithm 2.

Because probe-coordinate success does not by itself guarantee downstream generation quality, we also conduct a blinded LLM-as-judge evaluation of open-ended steering outputs in Appendix B.8. We use 100 frozen neutral movie prompts with three random seeds, yielding 300 prompt–seed cases. The judge first filters invalid outputs and then scores valid generations on target sentiment strength, output quality, naturalness, and overall steering success. We compare RAPTOR with GCS under the same GCAV-based strength calibration; xRFM is excluded because it is nonlinear and does not support the same closed-form calibration. On the fully valid subset, RAPTOR achieves stronger target sentiment control than GCS while maintaining comparable output quality and naturalness.

4 Mechanistic Analysis

Motivation. Few-shot probing often operates in a proportional high-dimensional regime where the layer representation dimension p can be comparable to, or exceed, the number of labeled examples n. In this regime, empirical risk minimization may enter a nearly separable phase, unregularized logistic regression can become ill-posed, and classical fixed-p theory can be inaccurate [28, 29, 30]. Ridge-regularized logistic regression is therefore a natural default: it restores well-posedness and introduces a single interpretable knob λ whose effect can be characterized sharply using tools from precise high-dimensional analysis, including AMP and CGMT-based frameworks [31, 32, 33, 34]. Mechanistically, λ controls the decomposition of the learned direction into signal-aligned and orthogonal components, motivating our use of λ as a unified control parameter for accuracy and directional stability.

Main results. We analyze ridge logistic regression under a Gaussian teacher–student model in the proportional regime $n, p \to \infty$ with $n/p \to \delta$. Using CGMT [32, 33], we obtain a deterministic fixed-point characterization in terms of order parameters $(\bar{\alpha}, \bar{\sigma})$ and an auxiliary scalar $\bar{\gamma}$. This yields a scalar expression for the limiting test accuracy and clarifies how λ shapes the signal–orthogonal decomposition of the learned direction. Appendix A gives a self-contained derivation; after matching notation, our fixed-point equations coincide with the ridge-logistic specialization of [34].

The Gaussian teacher–student model is not intended as an exact generative model for LLM layer representations. Rather, it is a tractable model for isolating the high-dimensional mechanism by which ridge regularization changes the learned direction. This mechanism predicts qualitative regularization trends, including non-monotonic dependence on λ, and is complementary to proportional-limit analyses of separable linear classification [29, 30]. It also predicts ratio-controlled behavior: at fixed representation dimension p, changing the number of examples n should primarily affect performance through the aspect ratio $\delta = n/p$. Our real-embedding subsampling experiment confirms this structural prediction: as n varies, RAPTOR's accuracy is largely governed by δ rather than the absolute scale of n.

This interpretation is consistent with a broader universality perspective in high-dimensional statistics, where proportional-limit phenomena for convex estimators can persist beyond exact Gaussian designs under suitable regularity and moment-matching conditions [35, 36]. In our setting, the practical role of the theory is explanatory rather than prescriptive: to explain why varying λ changes accuracy and directional stability, and why the resulting proportional trend remains visible on real LLM layer representations.

4.1 Model and proportional regime

We observe i.i.d. data $(x_i, y_i)_{i=1}^n$ with $x_i \in \mathbb{R}^p$ and labels $y_i \in \{\pm 1\}$. Assume the Gaussian design

$$x_i \sim \mathcal{N}\left(0, \frac{1}{p} I_p\right), \qquad i = 1, \ldots, n, \tag{7}$$

and a logistic teacher with parameter $\beta^\star \in \mathbb{R}^p$ satisfying $\|\beta^\star\|_2^2 = p\kappa^2$ for a constant signal level $\kappa > 0$:

$$\mathbb{P}(y_i = +1 \mid x_i) = \sigma(x_i^\top \beta^\star), \quad \mathbb{P}(y_i = -1 \mid x_i) = \sigma(-x_i^\top \beta^\star), \quad \sigma(t) = \frac{1}{1 + e^{-t}}. \tag{8}$$

We study the proportional limit $n, p \to \infty$ with $\delta := \lim_{p\to\infty} \frac{n}{p} \in (0, \infty)$.

4.2 Ridge logistic regression

Let $\ell(y, t) = \log(1 + e^{-yt})$ and $\rho(t) = \log(1 + e^t)$. Ridge logistic regression solves

$$\hat{\beta} \in \arg\min_{\beta\in\mathbb{R}^p} \Big\{ \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, x_i^\top \beta\big) + \frac{\lambda}{2p} \|\beta\|_2^2 \Big\}, \qquad \lambda > 0. \tag{9}$$

A convenient rescaling for the analysis is $z := \beta/\sqrt{p}$, so that $x_i^\top \beta$ becomes a standard Gaussian bilinear form. The key geometric quantities of the optimizer are its *alignment* with the teacher direction and its *orthogonal energy*:

$$\alpha = \langle \hat{z},\, v \rangle, \quad \sigma = \|\hat{z} - \alpha v\|_2, \quad v = \beta^\star / \|\beta^\star\|_2. \tag{10}$$

4.3 Fixed-point characterization

To state the proportional-limit characterization compactly, we introduce the proximal operator of ρ purely as notation (no algorithmic use is needed). Define the proximal map of ρ by

$$\eta_\gamma(u) := \mathrm{prox}_{\gamma\rho}(u) = \arg\min_{t\in\mathbb{R}} \Big\{ \rho(t) + \frac{1}{2\gamma}(t - u)^2 \Big\}, \qquad \gamma > 0. \tag{11}$$

Equivalently, $\eta_\gamma(u)$ is the unique solution of $\eta_\gamma(u) + \gamma\sigma\big(\eta_\gamma(u)\big) = u$.

Theorem 4.1 (Ridge logistic regression in the proportional regime; adapted from [34])**.** *Assume the model above with $n, p \to \infty$ and $n/p \to \delta \in (0, \infty)$, and fix $\lambda > 0$. Suppose that the following fixed-point system has a unique solution $(\bar{\alpha}, \bar{\sigma}, \bar{\gamma}) \in \mathbb{R} \times (0, \infty)^2$. Then the random pair (α, σ) defined above converges in probability to $(\bar{\alpha}, \bar{\sigma})$, where the limiting parameters satisfy*

$$1 = \frac{2\delta}{\bar{\sigma}^2}\, \mathbb{E}\Big[\sigma(-\kappa Z_1)\, \big(V - \eta_{\bar{\gamma}}(V)\big)^2\Big] \tag{12}$$

$$\frac{\bar{\alpha}}{\delta} = -2\kappa\, \mathbb{E}\Big[\sigma(-\kappa Z_1)\big(1 - \sigma(-\kappa Z_1)\big)\, \eta_{\bar{\gamma}}(V)\Big] \tag{13}$$

$$1 - \frac{1}{\delta} + \bar{\gamma}\lambda = \mathbb{E}\left[\frac{2\,\sigma(-\kappa Z_1)}{1 + \bar{\gamma}\,\sigma\big(\eta_{\bar{\gamma}}(V)\big)\big(1 - \sigma\big(\eta_{\bar{\gamma}}(V)\big)\big)}\right] \tag{14}$$

where $Z_1, Z_2 \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$ and $V = \bar{\alpha} Z_1 + \bar{\sigma} Z_2$.

4.4 Deterministic limit of test accuracy

Let (x, y) be an independent test pair from the same teacher model. Consider the zero-threshold classifier $\hat{y}(x) = \mathrm{sign}(x^\top \hat{\beta})$. Define the teacher score $U := x^\top \beta^\star$ and the learned score $S := x^\top \hat{\beta}$. Gaussian rotational invariance and Theorem 4.1 yield the following joint limit:

$$(U, S) \xrightarrow{d} \big(\kappa Z, \bar{\alpha} Z + \bar{\sigma} W\big), \qquad Z, W \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1). \tag{15}$$

Proposition 4.2 (Asymptotic test accuracy)**.** *Under the conditions of Theorem 4.1, the out-of-sample classification accuracy converges to*

$$\mathrm{Acc}(\delta, \lambda, \kappa) = \mathbb{E}_Z\Big[\, \sigma(\kappa Z)\, \Phi\Big(\frac{\bar{\alpha}}{\bar{\sigma}} Z\Big) + \sigma(-\kappa Z)\, \Big(1 - \Phi\Big(\frac{\bar{\alpha}}{\bar{\sigma}} Z\Big)\Big)\, \Big] \tag{16}$$

where $Z \sim \mathcal{N}(0, 1)$ and $(\bar{\alpha}, \bar{\sigma})$ are the first two components of the unique fixed-point solution specified in Theorem 4.1.

How λ affects directional stability Our robustness metric is *directional stability*, defined as the mean absolute pairwise cosine similarity between concept vectors learned across ablation runs. The fixed-point parameters $(\bar{\alpha}, \bar{\sigma})$ admit a simple geometric interpretation.

Defining $\hat{v} = \hat{z}/\|\hat{z}\|_2$, the orthogonal decomposition in Equation 10 and Theorem 4.1 give

$$\langle \hat{v}, v \rangle = \frac{\alpha}{\sqrt{\alpha^2 + \sigma^2}} \xrightarrow{P} \frac{\bar{\alpha}}{\sqrt{\bar{\alpha}^2 + \bar{\sigma}^2}}. \tag{17}$$

Under a stylized high-dimensional approximation in which two probes trained on independent subsamples yield orthogonal noise components, the cosine similarity between the resulting directions concentrates as

$$\cos(\hat{v}^{(1)}, \hat{v}^{(2)}) \approx \frac{\bar{\alpha}^2}{\bar{\alpha}^2 + \bar{\sigma}^2}. \tag{18}$$

Thus, increasing the signal component or suppressing orthogonal energy improves directional stability, providing a direct link between regularization and robustness.

4.5 Validating high-dimensional structure on real embeddings

We test whether the proportional-theory prediction that accuracy is governed by the aspect ratio $\delta = n/p$ is visible in real LLM embeddings. For each fixed model–dataset–layer tuple, we vary n by stratified subsampling (6 fractions, 5 seeds) at fixed representation dimension p, and compare Raptor's held-out accuracy $\mathrm{Acc}_{\text{true}}$ with a theory-inspired predictor $\mathrm{Acc}_{\text{pred}}$; details are in Appendix B.9. Across 12 settings (2 models $\times$ 3 datasets $\times$ 2 layers), $\mathrm{Acc}_{\text{pred}}$ tracks the δ sweep well, with median Spearman 0.86 and median Pearson 0.90 (best over the ridge grid); full results are reported in Table 7. This suggests that the proportional theory captures the dominant accuracy trend induced by changing n, despite the non-Gaussian and correlated structure of real LLM embeddings.

5 Discussion and Conclusion

In this work, we revisited the foundations of linear probing for inference-time intervention. While the literature often treats logistic regression as a static baseline, our findings demonstrate that its behavior in modern probing settings is heavily dependent on regularization and protocol choices. We introduce RAPTOR to formalize this insight: by reducing the probe design to a single, validation-selected ridge parameter λ, we achieve a minimal yet highly effective standard for concept extraction.

Empirically, RAPTOR challenges the assumption that accurate concept probing requires complex estimators. Across our benchmark, it achieves competitive accuracy while offering greater directional stability than xRFM under the evaluated ablations. This establishes a crucial practical takeaway: a rigorously tuned ridge-logistic probe serves as a formidable reference point for evaluating more complex estimators in activation steering.

To ground these empirical successes, we complemented our benchmark with a high-dimensional theoretical analysis of ridge logistic regression. Although our stylized Gaussian teacher–student model simplifies the complex distribution of real LLM representations, it provides analytical insight into why regularization is essential for recovering stable directions in high-dimensional spaces. Future work can extend this theoretical framework to encompass more realistic feature dependencies, thereby bridging the gap between statistical theory and the practical dynamics of LLMs.

Broader impacts. RAPTOR is intended to support analysis and controllability of language-model representations by making concept-vector estimation more accurate, stable, and efficient. The same capability could also be misused to manipulate model behavior or amplify undesirable concepts, so downstream deployment of steering methods should include task-specific safety evaluation and appropriate monitoring.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their helpful feedback. We acknowledge the computing resources and staff support of Quest, Northwestern University's high-performance computing facility.

The authors declare no competing interests.

References

[1] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

[2] Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872. Association for Computational Linguistics, 2017.

[3] John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, 2019.

[4] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601. Association for Computational Linguistics, 2019.

[5] Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2020.

[6] Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. GeDi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[7] Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online, August 2021. Association for Computational Linguistics.

[8] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2023.

[9] Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[10] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics.

[11] Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[12] Abhilasha Ravichander, Yonatan Belinkov, and Eduard Hovy. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3363–3377, Online, April 2021. Association for Computational Linguistics.

[13] Isha Agarwal, Saharsha Navani, and Fazl Barez. Context matters: Analyzing the generalizability of linear probing and steering across diverse scenarios. In *NeurIPS 2025 Workshop on Mechanistic Interpretability*, 2025.

[14] Daniel Chee Hian Tan, David Chanin, Aengus Lynch, Adrià Garriga-Alonso, Dimitrios Kanoulas, Brooks Paige, and Robert Kirk. Analyzing the generalization and reliability of steering vectors. *arXiv preprint arXiv:2407.12404*, 2024.

[15] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, March 2022.

[16] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

[17] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256. Association for Computational Linguistics, 2020.

[18] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 2021.

[19] Haiyan Zhao, Heng Zhao, Bo Shen, Ali Payani, Fan Yang, and Mengnan Du. Beyond single concept vector: Modeling concept subspace in LLMs with gaussian distribution. In *International Conference on Learning Representations*, 2025.

[20] Daniel Beaglehole, Adityanarayanan Radhakrishnan, Enric Boix-Adserà, and Mikhail Belkin. Aggregate and conquer: Detecting and steering LLM concepts by combining nonlinear predictors over multiple layers. *arXiv preprint arXiv:2502.03708*, 2025.

[21] Hanyu Zhang, Xiting Wang, Chengao Li, Xiang Ao, and Qing He. Controlling large language models through concept activation vectors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25851–25859, 2025.

[22] Zhihao Xu, Ruixuan Huang, Changyu Chen, and Xiting Wang. Uncovering safety risks of large language models through concept activation vector. *Advances in Neural Information Processing Systems*, 37:116743–116782, 2024.

[23] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, 2013. Association for Computational Linguistics.

[24] Mingyu Jin, Qinkai Yu, Jingyuan Huang, Qingcheng Zeng, Zhenting Wang, Wenyue Hua, Haiyan Zhao, Kai Mei, Yanda Meng, Kaize Ding, Fan Yang, Mengnan Du, and Yongfeng Zhang. Exploring concept depth: How large language models acquire knowledge and concept at different layers? In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 558–573. Association for Computational Linguistics, 2025.

[25] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

[26] Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. HateXplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14867–14875, 2021.

[27] Rishabh Misra and Prahal Arora. Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18, 2023.

[28] Pragya Sur and Emmanuel J. Candès. A modern maximum-likelihood theory for high-dimensional logistic regression. *Proceedings of the National Academy of Sciences*, 116(29):14516–14525, 2019.

[29] Andrea Montanari, Feng Ruan, Youngtak Sohn, and Jun Yan. The generalization error of max-margin linear classifiers: Benign overfitting and high dimensional asymptotics in the overparametrized regime. *The Annals of Statistics*, 53(2):822–853, 2025.

[30] Zeyu Deng, Abla Kammoun, and Christos Thrampoulidis. A model of double descent for high-dimensional binary linear classification. *Information and Inference: A Journal of the IMA*, 11(2):435–495, 2022.

[31] David Donoho and Andrea Montanari. High dimensional robust M-estimation: asymptotic variance via approximate message passing. *Probability Theory and Related Fields*, 166(3-4):935–969, 2016.

[32] Christos Thrampoulidis, Samet Oymak, and Babak Hassibi. Regularized linear regression: A precise analysis of the estimation error. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 1683–1709, Paris, France, July 2015. PMLR.

[33] Christos Thrampoulidis, Ehsan Abbasi, and Babak Hassibi. Precise error analysis of regularized m-estimators in high dimensions. *IEEE Transactions on Information Theory*, 64(8):5592–5628, 2018.

[34] Fariborz Salehi, Ehsan Abbasi, and Babak Hassibi. The impact of regularization on high-dimensional logistic regression. In *Advances in Neural Information Processing Systems*, 2019.

[35] Samet Oymak and Joel A. Tropp. Universality laws for randomized dimension reduction, with applications. *Information and Inference: A Journal of the IMA*, 7(3):337–446, 2018.

[36] Ashkan Panahi and Babak Hassibi. A universal analysis of large-scale regularized least squares solutions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

A CGMT derivation of Theorem 4.1

This appendix gives a CGMT-based derivation of Theorem 4.1. After matching the signal-alignment normalization, the resulting fixed-point equations are equivalent to those in [34]. The asymptotic optimizer-convergence step uses that result, with its uniqueness assumption stated explicitly below. The main text states the final fixed-point system (Theorem 4.1) and the limiting test accuracy (Proposition 4.2); here we give a CGMT-based route from the ERM (9) to the deterministic characterization.

A.1 Model, estimator, and scaling

A.1.1 Data model and proportional regime

We observe a feature matrix and labels

$$\mathbf{X} := \begin{bmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix} \in \mathbb{R}^{n\times p}, \qquad \mathbf{y} := (y_1, \dots, y_n) \in \{\pm 1\}^n.$$

We work in the *proportional* asymptotic regime

$$\delta \in (0, \infty), \qquad n, p \to \infty \text{ with } \frac{n}{p} \to \delta.$$

The three scalar parameters that will index the final accuracy are:

- $\delta = \lim n/p$: limiting sample-to-dimension ratio,
- $\lambda > 0$: ridge strength,
- $\kappa > 0$: signal strength of the teacher.

Assume

$$\mathbf{X} = \frac{1}{\sqrt{p}}\mathbf{M}, \qquad \mathbf{M} \in \mathbb{R}^{n\times p} \text{ has i.i.d. } \mathcal{N}(0,1) \text{ entries.}$$

Equivalently, the rows satisfy $\mathbf{x}_i \sim \mathcal{N}(0, \mathbf{I}_p/p)$ independently.

Fix a *true parameter* $\boldsymbol{\beta}^\star \in \mathbb{R}^p$ with

$$\|\boldsymbol{\beta}^\star\|^2 = p\kappa^2, \qquad \kappa > 0 \text{ constant.}$$

Let $\mathbf{v} := \boldsymbol{\beta}^\star/\|\boldsymbol{\beta}^\star\|$ so that $\|\mathbf{v}\| = 1$ and $\boldsymbol{\beta}^\star = \kappa\sqrt{p}\,\mathbf{v}$. Define the teacher score for sample i:

$$U_i := \mathbf{x}_i^\top \boldsymbol{\beta}^\star = \frac{1}{\sqrt{p}}\mathbf{M}_{i:}\boldsymbol{\beta}^\star = \kappa\,(\mathbf{M}\mathbf{v})_i.$$

Conditioned on $\mathbf{X}$ (equivalently on $\mathbf{M}$), the labels are independent with

$$\mathbb{P}(y_i = +1 \mid \mathbf{X}) = \sigma(U_i), \qquad \mathbb{P}(y_i = -1 \mid \mathbf{X}) = \sigma(-U_i),$$

where $\sigma(u) := 1/(1+e^{-u})$.

A.1.2 Estimator (ridge logistic regression)

Define the logistic loss $\ell(y,t) := \log(1+e^{-yt}) = \rho(-yt)$ and the log-partition function

$$\rho(u) := \log(1+e^u).$$

We consider the ridge-logistic scaling used in [34]:

$$\hat{\boldsymbol{\beta}} \in \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^p} \Big\{ \frac{1}{n}\sum_{i=1}^n \ell\big(y_i, \mathbf{x}_i^\top\boldsymbol{\beta}\big) + \frac{\lambda}{2p}\|\boldsymbol{\beta}\|^2 \Big\}, \qquad \lambda > 0. \tag{19}$$

Multiplying the objective by n (which does not change the minimizer) gives the equivalent form

$$\hat{\boldsymbol{\beta}} \in \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^p} \Big\{ \sum_{i=1}^n \ell\big(y_i, \mathbf{x}_i^\top\boldsymbol{\beta}\big) + \frac{n\lambda}{2p}\|\boldsymbol{\beta}\|^2 \Big\}.$$

A.1.3 Convenient rescaling

Introduce the scaled variable

$$\mathbf{z} := \frac{\boldsymbol{\beta}}{\sqrt{p}} \in \mathbb{R}^p \quad \Longleftrightarrow \quad \boldsymbol{\beta} = \sqrt{p}\,\mathbf{z}.$$

Then $\mathbf{x}_i^\top \boldsymbol{\beta} = (\mathbf{M}_{i:}\mathbf{z})$ and $\|\boldsymbol{\beta}\|^2 = p\|\mathbf{z}\|^2$. Substituting into the unnormalized objective gives

$$\hat{\mathbf{z}} \in \operatorname*{arg\,min}_{\mathbf{z}\in\mathbb{R}^p} \Big\{ \sum_{i=1}^n \rho\big(-y_i(\mathbf{M}_{i:}\mathbf{z})\big) + \frac{n\lambda}{2}\|\mathbf{z}\|^2 \Big\}. \tag{20}$$

We will characterize the limiting signal-aligned and orthogonal components of $\hat{\mathbf{z}}$, then translate back to $\hat{\boldsymbol{\beta}} = \sqrt{p}\,\hat{\mathbf{z}}$.

A.2 From ERM to a convex min–max (Primary Optimization)

Lemma A.1 (Convex conjugate of $\rho(u) = \log(1+e^u)$)**.** *Let $\rho(u) = \log(1+e^u)$. Its Fenchel conjugate ρ^* is*

$$\rho^*(s) = \begin{cases} s\log s + (1-s)\log(1-s), & s \in (0,1), \\ 0, & s \in \{0,1\}, \\ +\infty, & \text{otherwise}. \end{cases}$$

Moreover, for all $u \in \mathbb{R}$,

$$\rho(u) = \max_{s\in[0,1]} \{\, s\,u - \rho^*(s)\,\}. \tag{21}$$

The maximizer is $s^\star = \sigma(u)$.

Starting from (20), apply Lemma A.1 coordinate-wise with $u = -y_i(\mathbf{M}_{i:}\mathbf{z})$:

$$\rho\big(-y_i\mathbf{M}_{i:}\mathbf{z}\big) = \max_{w_i\in[0,1]} \Big\{ -w_i\,y_i\,(\mathbf{M}_{i:}\mathbf{z}) - \rho^*(w_i) \Big\}.$$

Let $\mathbf{w} = (w_1,\dots,w_n) \in [0,1]^n$. Plugging into (20) and exchanging the finite sum with the maximization gives the saddle formulation

$$\boxed{\Phi(\mathbf{M},\mathbf{y}) := \min_{\mathbf{z}\in\mathbb{R}^p} \max_{\mathbf{w}\in[0,1]^n} \Big\{ -\mathbf{w}^\top \operatorname{diag}(\mathbf{y})\mathbf{M}\mathbf{z} - \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}\|\mathbf{z}\|^2 \Big\}} \tag{22}$$

The vector $\mathbf{w} \in [0,1]^n$ is the Fenchel dual variable arising from (21); there is no additional sign constraint tied to $\mathbf{y}$.

Why an independence reduction is necessary. The Gaussian matrix $\mathbf{M}$ and the labels $\mathbf{y}$ are dependent: $\mathbf{y}$ is generated from the teacher score $\mathbf{M}\mathbf{v}$. To apply CGMT, we isolate *all* dependence into a single Gaussian vector and keep an independent Gaussian matrix in the remaining bilinear term.

A.3 Isolating label dependence via orthogonal decomposition

A.3.1 Gaussian object that generates y

Define the signal-direction score vector

$$\mathbf{u} := \mathbf{M}\mathbf{v} \in \mathbb{R}^n.$$

Since $\mathbf{M}$ has i.i.d. $\mathcal{N}(0,1)$ entries and $\|\mathbf{v}\| = 1$, we have $\mathbf{u} \sim \mathcal{N}(0,\mathbf{I}_n)$. Moreover,

$$\mathbb{P}(y_i = +1 \mid u_i) = \sigma(\kappa u_i), \qquad \mathbb{P}(y_i = -1 \mid u_i) = \sigma(-\kappa u_i),$$

so the entire dependence of $\mathbf{y}$ on $\mathbf{M}$ is through the coordinates $\{u_i\}_{i=1}^n$.

Lemma A.2 (Gaussian orthogonal decomposition)**.** *Let $\mathbf{M} \in \mathbb{R}^{n\times p}$ have i.i.d. $\mathcal{N}(0,1)$ entries and let $\mathbf{v} \in \mathbb{R}^p$ be deterministic with $\|\mathbf{v}\| = 1$. Set $\mathbf{u} := \mathbf{M}\mathbf{v}$. Then there exists a random matrix $\mathbf{M}_\perp$ such that*

$$\mathbf{M} = \mathbf{u}\,\mathbf{v}^\top + \mathbf{M}_\perp, \qquad \mathbf{M}_\perp\mathbf{v} = \mathbf{0},$$

and $\mathbf{M}_\perp$ is independent of $\mathbf{u}$. In an orthonormal basis of $\mathbf{v}^\perp$, the coordinates of $\mathbf{M}_\perp$ are i.i.d. $\mathcal{N}(0,1)$.

A.3.2 Decompose both the matrix and the primal variable

Using Lemma A.2, write

$$\mathbf{M} = \mathbf{u}\,\mathbf{v}^\top + \mathbf{M}_\perp, \qquad \mathbf{M}_\perp \mathbf{v} = \mathbf{0}. \tag{23}$$

Decompose the optimization variable similarly:

$$\mathbf{z} = \alpha\,\mathbf{v} + \mathbf{z}_\perp, \qquad \alpha \in \mathbb{R},\ \mathbf{z}_\perp \in \mathbf{v}^\perp. \tag{24}$$

Then $\|\mathbf{z}\|^2 = \alpha^2 + \|\mathbf{z}_\perp\|^2$ and

$$\mathbf{M}\mathbf{z} = (\mathbf{u}\,\mathbf{v}^\top)\mathbf{z} + \mathbf{M}_\perp \mathbf{z}_\perp = \alpha\,\mathbf{u} + \mathbf{M}_\perp \mathbf{z}_\perp.$$

Plugging (23)–(24) into (22) gives

$$\begin{aligned}\Phi = \min_{\alpha\in\mathbb{R},\ \mathbf{z}_\perp\in\mathbf{v}^\perp} \max_{\mathbf{w}\in[0,1]^n} \Big\{ &- \alpha\,\mathbf{w}^\top \operatorname{diag}(\mathbf{y})\mathbf{u} - \mathbf{w}^\top \operatorname{diag}(\mathbf{y})\mathbf{M}_\perp \mathbf{z}_\perp \\ &- \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}\big(\alpha^2 + \|\mathbf{z}_\perp\|^2\big)\Big\}.\end{aligned} \tag{25}$$

Define the two objects that remain throughout the CGMT step:

$$\mathbf{s} := \operatorname{diag}(\mathbf{y})\mathbf{u} \in \mathbb{R}^n, \qquad \mathbf{A} := \operatorname{diag}(\mathbf{y})\mathbf{M}_\perp|_{\mathbf{v}^\perp}.$$

We represent the second operator in a fixed orthonormal basis of $\mathbf{v}^\perp$, so that $\mathbf{A} \in \mathbb{R}^{n\times(p-1)}$. When multiplying by $\mathbf{A}$, we identify $\mathbf{z}_\perp$ with its coordinates in this basis. Then

$$\begin{aligned}\Phi = \min_{\alpha\in\mathbb{R},\ \mathbf{z}_\perp\in\mathbf{v}^\perp} \max_{\mathbf{w}\in[0,1]^n} \Big\{ &- \alpha\,\mathbf{w}^\top \mathbf{s} - \mathbf{w}^\top \mathbf{A}\mathbf{z}_\perp \\ &- \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}\big(\alpha^2 + \|\mathbf{z}_\perp\|^2\big)\Big\}.\end{aligned} \tag{26}$$

Conditional on $(\mathbf{u}, \mathbf{y})$, and hence also on $\mathbf{s}$, the entries of $\mathbf{A}$ in this basis are i.i.d. standard Gaussian. Its conditional distribution does not depend on $(\mathbf{u}, \mathbf{y})$, so $\mathbf{A}$ is independent of $\mathbf{s}$. Indeed, $\mathbf{M}_\perp$ is independent of $(\mathbf{u}, \mathbf{y})$, since the labels are generated from $\mathbf{u}$ alone, and multiplying its rows by ± 1 preserves its Gaussian distribution.

A.4 CGMT reduction: auxiliary optimization

A.4.1 CGMT form used

We invoke the convex Gaussian min–max theorem (CGMT) in the standard bilinear form; see, e.g., [32, 33].

Theorem A.3 (CGMT, specialized form)**.** *Let $\mathbf{A} \in \mathbb{R}^{n\times(p-1)}$, $\mathbf{g} \in \mathbb{R}^n$, and $\mathbf{h} \in \mathbb{R}^{p-1}$ have mutually independent standard Gaussian entries. Let $\mathcal{S} \subset \mathbb{R} \times \mathbb{R}^{p-1}$ and $\mathcal{T} \subset \mathbb{R}^n$ be nonempty compact convex sets, and let $\Psi(\alpha, \mathbf{z}_\perp, \mathbf{w})$ be deterministic and continuous, convex in $(\alpha, \mathbf{z}_\perp)$, and concave in $\mathbf{w}$. Define*

$$\begin{aligned}\Phi_{\mathrm{PO}} &:= \min_{(\alpha,\mathbf{z}_\perp)\in\mathcal{S}} \max_{\mathbf{w}\in\mathcal{T}} \big\{\mathbf{w}^\top \mathbf{A}\mathbf{z}_\perp + \Psi(\alpha, \mathbf{z}_\perp, \mathbf{w})\big\}, \\ \Phi_{\mathrm{AO}} &:= \min_{(\alpha,\mathbf{z}_\perp)\in\mathcal{S}} \max_{\mathbf{w}\in\mathcal{T}} \big\{\|\mathbf{z}_\perp\|\,\mathbf{g}^\top \mathbf{w} + \|\mathbf{w}\|\,\mathbf{h}^\top \mathbf{z}_\perp + \Psi(\alpha, \mathbf{z}_\perp, \mathbf{w})\big\}.\end{aligned}$$

For every $\mu \in \mathbb{R}$ and $t > 0$,

$$\mathbb{P}\big(|\Phi_{\mathrm{PO}} - \mu| > t\big) \le 2\mathbb{P}\big(|\Phi_{\mathrm{AO}} - \mu| \ge t\big).$$

The usual strict-gap comparison for a restricted minimizing set transfers concentration of optimizer summaries from the AO to the PO.

Here α is an auxiliary minimizing coordinate that enters only through Ψ. This form follows from the standard CGMT by introducing a positive scale ε, appending an independent Gaussian column acting on $\varepsilon\alpha$, and letting $\varepsilon \downarrow 0$. Compactness makes both optimization values continuous under this limit.

A.4.2 Apply CGMT to (26)

In (26), conditional on $\mathbf{s}$, the only random Gaussian object is $\mathbf{A}$ and it appears only in the bilinear term $-\mathbf{w}^\top \mathbf{A}\mathbf{z}_\perp$. The compactness requirement does not change the optimizer: the PO objective is at least $n\lambda\|\mathbf{z}\|^2/2$, while its value at $\mathbf{z} = \mathbf{0}$ is $n \log 2$. Thus the primal domain may be restricted to a closed ball of any fixed radius greater than $\sqrt{2\log 2/\lambda}$. The same bound holds for the AO below: choosing $\mathbf{w} = \mathbf{0}$ gives the quadratic lower bound, and the value at $(\alpha, \mathbf{z}_\perp) = (0, \mathbf{0})$ is $n \log 2$. We omit this inactive ball constraint in the following displays.

Absorb the minus sign by redefining $\mathbf{A} \leftarrow -\mathbf{A}$ (distribution unchanged). CGMT then yields the auxiliary optimization

$$\begin{aligned}\phi = \min_{\alpha\in\mathbb{R},\ \mathbf{z}_\perp\in\mathbf{v}^\perp} \max_{\mathbf{w}\in[0,1]^n} \Big\{ &-\alpha\,\mathbf{w}^\top\mathbf{s} + \|\mathbf{z}_\perp\|\,\mathbf{g}^\top\mathbf{w} + \|\mathbf{w}\|\,\mathbf{h}^\top\mathbf{z}_\perp \\ &- \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}\big(\alpha^2 + \|\mathbf{z}_\perp\|^2\big)\Big\}.\end{aligned} \tag{27}$$

Here $\mathbf{g} \in \mathbb{R}^n$ and $\mathbf{h} \in \mathbb{R}^{p-1}$ are independent standard Gaussian vectors, independent of $(\mathbf{u}, \mathbf{y})$.

A.5 Scalarization and decoupling of the auxiliary optimization

A.5.1 Eliminate the direction of $\mathbf{z}_\perp$

Fix α, $\mathbf{w}$, and let $r := \|\mathbf{z}_\perp\| \geq 0$. The only term depending on the direction of $\mathbf{z}_\perp$ is $\|\mathbf{w}\|\,\mathbf{h}^\top\mathbf{z}_\perp$. Minimizing over all $\mathbf{z}_\perp \in \mathbf{v}^\perp$ with $\|\mathbf{z}_\perp\| = r$ yields

$$\min_{\|\mathbf{z}_\perp\|=r} \|\mathbf{w}\|\,\mathbf{h}^\top\mathbf{z}_\perp = -\|\mathbf{w}\|\,r\,\|\mathbf{h}\|.$$

For $\mathbf{h} \neq \mathbf{0}$, the minimizing direction is $-\mathbf{h}/\|\mathbf{h}\|$, independently of $\mathbf{w}$; if $\mathbf{h} = \mathbf{0}$, every direction gives the same value. Thus the same direction minimizes the expression for every dual vector, and (27) becomes

$$\phi = \min_{\alpha\in\mathbb{R},\ r\geq 0} \max_{\mathbf{w}\in[0,1]^n} \Big\{ -\alpha\,\mathbf{w}^\top\mathbf{s} + r\,\mathbf{g}^\top\mathbf{w} - r\,\|\mathbf{h}\|\,\|\mathbf{w}\| - \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}(\alpha^2 + r^2)\Big\}. \tag{28}$$

A.5.2 Two variational identities used to decouple norms and squares

Lemma A.4 (Linearize a norm)**.** *For any $a \geq 0$ and any vector $\mathbf{w} \in \mathbb{R}^n$,*

$$-a\|\mathbf{w}\| = \sup_{\tau>0}\left\{-\frac{\tau}{2}\|\mathbf{w}\|^2 - \frac{a^2}{2\tau}\right\}.$$

Lemma A.5 (Linearize a negative square)**.** *For any $c > 0$ and scalar $u \in \mathbb{R}$,*

$$-\frac{u^2}{2c} = \min_{\gamma\in\mathbb{R}}\left\{\frac{c}{2}\gamma^2 - \gamma u\right\}.$$

A.5.3 Decouple $-\|\mathbf{w}\|$

Apply Lemma A.4 to the coupling term with $a = r\|\mathbf{h}\|$:

$$-r\|\mathbf{h}\|\,\|\mathbf{w}\| = \sup_{\tau>0}\left\{-\frac{\tau}{2}\|\mathbf{w}\|^2 - \frac{r^2\|\mathbf{h}\|^2}{2\tau}\right\}.$$

Substituting into (28) gives

$$\begin{aligned}\phi = \min_{\alpha\in\mathbb{R},\ r\geq 0} \sup_{\tau>0} \max_{\mathbf{w}\in[0,1]^n} \Big\{ &-\alpha\,\mathbf{w}^\top\mathbf{s} + r\,\mathbf{g}^\top\mathbf{w} - \frac{\tau}{2}\|\mathbf{w}\|^2 \\ &- \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2}(\alpha^2 + r^2) - \frac{r^2\|\mathbf{h}\|^2}{2\tau}\Big\}.\end{aligned} \tag{29}$$

A.5.4 Eliminate α explicitly

For fixed r, the objective in (28) is convex and uniformly coercive in α, and concave in $\mathbf{w} \in [0,1]^n$. The convex–concave minimax equality therefore allows $\min_\alpha$ and $\max_{\mathbf{w}}$ to be exchanged in (28), before introducing τ. For fixed $(r, \mathbf{w})$, the α-dependent part is

$$-\alpha\, \mathbf{w}^\top \mathbf{s} + \frac{n\lambda}{2}\alpha^2,$$

whose minimum occurs at $\alpha^\star = (\mathbf{w}^\top \mathbf{s})/(n\lambda)$ and equals $-(\mathbf{w}^\top \mathbf{s})^2/(2n\lambda)$. Applying the norm identity after this elimination, and commuting the two suprema, gives

$$\phi = \min_{r\geq 0} \sup_{\tau>0} \max_{\mathbf{w}\in[0,1]^n} \Big\{ r\, \mathbf{g}^\top \mathbf{w} - \frac{\tau}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^n \rho^*(w_i) + \frac{n\lambda}{2} r^2 - \frac{r^2\|\mathbf{h}\|^2}{2\tau} - \frac{(\mathbf{w}^\top \mathbf{s})^2}{2n\lambda} \Big\}. \tag{30}$$

A.5.5 Decouple $(\mathbf{w}^\top \mathbf{s})^2$

Apply Lemma A.5 with $u = \mathbf{w}^\top \mathbf{s}$ and $c = n\lambda$, reintroducing the eliminated scalar α:

$$-\frac{(\mathbf{w}^\top \mathbf{s})^2}{2n\lambda} = \min_{\alpha\in\mathbb{R}} \left\{ \frac{n\lambda}{2}\alpha^2 - \alpha\, \mathbf{w}^\top \mathbf{s} \right\}.$$

For each fixed (r, τ), the resulting objective is convex and uniformly coercive in α and concave in the compact dual variable $\mathbf{w}$. The convex–concave minimax equality gives

$$\begin{aligned} \phi = \min_{r\geq 0} \sup_{\tau>0} \min_{\alpha\in\mathbb{R}} \max_{\mathbf{w}\in[0,1]^n} \Big\{ \sum_{i=1}^n \Big[(r g_i - \alpha s_i) w_i - \frac{\tau}{2} w_i^2 - \rho^*(w_i) \Big] \\ + \frac{n\lambda}{2} r^2 - \frac{r^2\|\mathbf{h}\|^2}{2\tau} + \frac{n\lambda}{2}\alpha^2 \Big\}. \end{aligned} \tag{31}$$

Now the maximization over $\mathbf{w}$ is coordinate-wise separable.

A.6 Coordinate-wise maximization and the logistic proximal map

Lemma A.6 (Prox definition)**.** *For a proper closed convex $f : \mathbb{R} \to (-\infty, +\infty]$ and $\gamma > 0$,*

$$\operatorname{prox}_{\gamma f}(v) := \arg\min_{u\in\mathbb{R}} \Big\{ f(u) + \frac{1}{2\gamma}(u - v)^2 \Big\}$$

Fix (r, τ, α) and define $a_i := r g_i - \alpha s_i$. For each coordinate we need

$$\max_{w\in[0,1]} \Big\{ a_i w - \frac{\tau}{2} w^2 - \rho^*(w) \Big\}$$

Completing the square gives

$$\begin{aligned} \max_{w\in[0,1]} \Big\{ a_i w - \frac{\tau}{2} w^2 - \rho^*(w) \Big\} &= \frac{a_i^2}{2\tau} \\ - \min_{w\in[0,1]} \Big\{ \rho^*(w) + \frac{\tau}{2}\Big(w - \frac{a_i}{\tau} \Big)^2 \Big\} & \end{aligned}$$

By Lemma A.6, the maximizer is

$$w_i^\star = \operatorname{prox}_{(1/\tau)\rho^*}\Big(\frac{a_i}{\tau} \Big)$$

Lemma A.7 (Moreau decomposition (scalar form))**.** *Let f be proper closed convex with conjugate $f^\star$. For any $\gamma > 0$ and $v \in \mathbb{R}$,*

$$\operatorname{prox}_{\gamma f}(v) + \gamma\, \operatorname{prox}_{f^\star/\gamma}\Big(\frac{v}{\gamma} \Big) = v$$

Equivalently,

$$\operatorname{prox}_{(1/\gamma) f^\star}(v) = v - \frac{1}{\gamma}\, \operatorname{prox}_{\gamma f}(\gamma v)$$

A.6.1 Convert prox of ρ^* into prox of ρ

Using Lemma A.7 with $f = \rho$ and $f^\star = \rho^*$ and $\gamma = \tau$,

$$\mathrm{prox}_{(1/\tau)\rho^*}(v) = v - \frac{1}{\tau}\,\mathrm{prox}_{\tau\rho}(\tau v)$$

Apply this with $v = a_i/\tau$ and define

$$\eta_i := \mathrm{prox}_{\tau\rho}(a_i)$$

Then

$$w_i^\star = \frac{a_i}{\tau} - \frac{1}{\tau}\,\eta_i \tag{32}$$

A.6.2 Implicit equation for the logistic proximal map

Since $\rho'(t) = \sigma(t)$, the first-order condition for $\eta = \mathrm{prox}_{\tau\rho}(a)$ is

$$0 = \rho'(\eta) + \frac{1}{\tau}(\eta - a) \quad \Longleftrightarrow \quad \boxed{\eta + \tau\sigma(\eta) = a}.$$

This has a unique solution for each (a, τ) because the left-hand side is strictly increasing in η and has range $\mathbb{R}$.

Define the Moreau envelope of ρ by

$$M_\rho(a;\tau) := \min_{u\in\mathbb{R}} \left\{ \rho(u) + \frac{(u-a)^2}{2\tau} \right\} = \max_{w\in[0,1]} \left\{ aw - \frac{\tau}{2}w^2 - \rho^*(w) \right\}.$$

Thus the coordinate-wise optimal value is $M_\rho(a_i;\tau)$.

A.7 Deterministic fixed-point characterization

Write the estimator in the teacher geometry:

$$\hat{\mathbf{z}} = \alpha\,\mathbf{v} + \mathbf{z}_\perp, \qquad \alpha = \langle \hat{\mathbf{z}}, \mathbf{v} \rangle, \qquad \sigma := \|\mathbf{z}_\perp\|.$$

Use σ for the radius r and γ for the proximal parameter τ in (29). Substituting the coordinate-wise optimal values gives

$$\frac{\phi}{n} = \min_{\alpha\in\mathbb{R},\ \sigma\geq 0} \sup_{\gamma>0} \left\{ \frac{\lambda}{2}(\alpha^2+\sigma^2) - \frac{\sigma^2\|\mathbf{h}\|^2}{2n\gamma} + \frac{1}{n}\sum_{i=1}^n M_\rho(-\alpha s_i + \sigma g_i;\gamma) \right\}.$$

Let Z_1, Z_2 be independent standard Gaussian variables and define

$$V := \alpha Z_1 + \sigma Z_2, \qquad q := \rho'(-\kappa Z_1).$$

For any integrable function f, Gaussian symmetry and the label model imply

$$\mathbb{E}\big[f(-\alpha y_i u_i + \sigma g_i)\big] = 2\mathbb{E}\big[\rho'(-\kappa Z_1) f(\alpha Z_1 + \sigma Z_2)\big].$$

Since $\|\mathbf{h}\|^2/n \to 1/\delta$, the pointwise population limit of the scalar objective is

$$F(\alpha,\sigma,\gamma) := \frac{\lambda}{2}(\alpha^2+\sigma^2) - \frac{\sigma^2}{2\delta\gamma} + 2\mathbb{E}\big[q\,M_\rho(V;\gamma)\big]. \tag{33}$$

For $\gamma > 0$, define

$$\eta_\gamma(t) := \mathrm{prox}_{\gamma\rho}(t), \qquad w_\gamma(t) := \rho'\big(\eta_\gamma(t)\big) = \frac{t - \eta_\gamma(t)}{\gamma}.$$

The proximal equation and its derivative give

$$\eta_\gamma(t) + \gamma\sigma\big(\eta_\gamma(t)\big) = t, \qquad w'_\gamma(t) = \frac{\rho''(\eta_\gamma(t))}{1 + \gamma\rho''(\eta_\gamma(t))}.$$

At an interior saddle point, differentiating (33) yields

$$0 = \lambda\alpha + 2\mathbb{E}\big[qZ_1 w_\gamma(V)\big],$$
$$0 = \lambda\sigma - \frac{\sigma}{\delta\gamma} + 2\mathbb{E}\big[qZ_2 w_\gamma(V)\big],$$
$$0 = \frac{\sigma^2}{2\delta\gamma^2} - \mathbb{E}\big[q\, w_\gamma(V)^2\big].$$

Gaussian integration by parts in Z_2, followed by division by $\sigma > 0$, gives

$$\lambda + 2\mathbb{E}\big[q\, w'_\gamma(V)\big] = \frac{1}{\delta\gamma}.$$

Integration by parts in Z_1 then gives

$$\frac{\alpha}{\delta\gamma} = 2\kappa\mathbb{E}\big[\rho''(-\kappa Z_1) w_\gamma(V)\big].$$

Because ρ'' is even, $\mathbb{E}[\rho''(-\kappa Z_1)V] = 0$. Using $\gamma w_\gamma(V) = V - \eta_\gamma(V)$ and $2\mathbb{E}[q] = 1$, these three stationarity equations reduce to (34)–(36) below.

Theorem A.8 (Fixed-point system for ridge logistic regression)**.** *Assume the Gaussian teacher–student logistic model specified above, with $n, p \to \infty$, $n/p \to \delta \in (0, \infty)$, $\kappa > 0$, and fixed $\lambda > 0$. Assume that* (34)–(36) *have a unique solution with $\bar\alpha \in \mathbb{R}$, $\bar\sigma > 0$, and $\bar\gamma > 0$. Then, for the estimator* (19)*,*

$$(\alpha, \sigma) = \big(\langle\hat{\mathbf{z}}, \mathbf{v}\rangle\,,\ \big\|\hat{\mathbf{z}} - \langle\hat{\mathbf{z}}, \mathbf{v}\rangle\,\mathbf{v}\big\|\big) \xrightarrow{\mathrm{P}} (\bar\alpha, \bar\sigma),$$

where the limiting parameters satisfy

$$1 = \frac{2\delta}{\bar\sigma^2}\,\mathbb{E}\Big[\rho'(-\kappa Z_1)\big(V - \eta_{\bar\gamma}(V)\big)^2\Big], \tag{34}$$

$$\frac{\bar\alpha}{\delta} = -2\kappa\,\mathbb{E}\Big[\rho''(-\kappa Z_1)\eta_{\bar\gamma}(V)\Big], \tag{35}$$

$$1 - \frac{1}{\delta} + \bar\gamma\lambda = \mathbb{E}\left[\frac{2\rho'(-\kappa Z_1)}{1 + \bar\gamma\rho''(\eta_{\bar\gamma}(V))}\right], \tag{36}$$

with $V = \bar\alpha Z_1 + \bar\sigma Z_2$, $\rho'(t) = \sigma(t)$, and $\rho''(t) = \sigma(t)(1 - \sigma(t))$.

Proof. The calculation above derives the fixed-point equations from the limiting scalar objective. For optimizer convergence, we invoke the ridge specialization in [34, Theorem 2], retaining its uniqueness assumption. The signal multiplier in that result equals $\bar\alpha/\kappa$ in our notation; $\bar\sigma$, $\bar\gamma$, and λ have the same normalization. The deterministic teacher direction causes no additional restriction: Gaussian design and the ridge penalty are rotationally invariant, so the teacher may be rotated to a vector with every coordinate equal to κ. Those constant coordinates satisfy the i.i.d. teacher-coordinate assumption using a point-mass distribution. The cited convergence result therefore gives the stated limits. □

A.8 Asymptotic test accuracy and its dependence on $(\delta, \lambda, \kappa)$

A.8.1 Joint Gaussian limit of test scores

Draw an independent test point $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I}_p/p)$ and a test label $y \in \{\pm 1\}$ generated by the same teacher:

$$\mathbb{P}(y = +1 \mid \mathbf{x}) = \sigma(\mathbf{x}^\top\boldsymbol{\beta}^\star), \qquad \mathbb{P}(y = -1 \mid \mathbf{x}) = \sigma(-\mathbf{x}^\top\boldsymbol{\beta}^\star).$$

Define the teacher score and learned score by

$$U := \mathbf{x}^\top\boldsymbol{\beta}^\star, \qquad S := \mathbf{x}^\top\hat{\boldsymbol{\beta}}.$$

Using $\hat{\boldsymbol{\beta}} = \sqrt{p}\,\hat{\mathbf{z}}$, $\boldsymbol{\beta}^\star = \kappa\sqrt{p}\,\mathbf{v}$, and the orthogonal decomposition of $\hat{\mathbf{z}}$, Gaussian rotational invariance gives the exact conditional distributional identity

$$(U, S) \mid (\mathbf{X}, \mathbf{y}) \overset{\mathrm{d}}{=} (\kappa Z, \alpha Z + \sigma W),$$

where Z, W are independent standard Gaussian variables, independent of the training data, and (α, σ) are fixed by those data. Consequently, Theorem A.8 gives the joint limit

$$(U, S) \xrightarrow{\mathrm{d}} (\kappa Z, \bar\alpha Z + \bar\sigma W).$$

A.8.2 Closed form for accuracy of the zero-threshold classifier

Consider the classifier $\hat{y} = \mathrm{sign}(S)$; ties have probability zero in the limiting model since $\bar{\sigma} > 0$. Let Acc denote the deterministic limit, in probability, of the conditional test accuracy $\mathbb{P}(\hat{y} = y \mid \mathbf{X}, \mathbf{y})$. Conditional on Z, the test label satisfies $\mathbb{P}(y = +1 \mid Z) = \sigma(\kappa Z)$ and is independent of W. The limiting learned score has distribution $\mathcal{N}(\bar{\alpha} Z, \bar{\sigma}^2)$ given Z. Hence

$$\mathrm{Acc} = \mathbb{E}_Z \left[\sigma(\kappa Z)\, \Phi\left(\frac{\bar{\alpha} Z}{\bar{\sigma}} \right) + \sigma(-\kappa Z) \left(1 - \Phi\left(\frac{\bar{\alpha} Z}{\bar{\sigma}} \right) \right) \right], \tag{37}$$

where Φ is the standard normal cumulative distribution function.

Corollary A.9 (Final performance characterization)**.** *Under the assumptions of Theorem A.8, the asymptotic test accuracy of ridge logistic regression is given by* (37)*.*

A.8.3 What are the independent variables, and how does Acc vary?

The accuracy is a deterministic function

$$\mathrm{Acc} = \mathrm{Acc}(\delta, \lambda, \kappa),$$

through the fixed-point solution $(\bar{\alpha}, \bar{\sigma})$ in Theorem 4.1. It is useful to summarize the dependence through the single *effective signal-to-noise ratio*

$$m := \frac{\bar{\alpha}}{\bar{\sigma}}. \tag{38}$$

Indeed, in (37) the learned classifier enters only through m (and the teacher enters through κ).

Interpretation via an effective margin m. For fixed κ, the accuracy expression (37) is increasing in an *effective margin* m on the relevant branch $\bar{\alpha} \geq 0$: larger m means the test score S is more aligned with the teacher score U and has less orthogonal noise relative to its signal component. Two useful limits are:

- If $m = 0$, then S is independent of y and $\mathrm{Acc} = 1/2$.
- As $m \to \infty$ at fixed κ, $\mathbb{P}(\mathrm{sign}(S) = \mathrm{sign}(U)) \to 1$ and $\mathrm{Acc} \to \mathbb{E}_Z\left[\sigma(\kappa |Z|)\right]$ (the Bayes-optimal accuracy under the logistic teacher).

B Experimental details

B.1 Raptor algorithm details

Details are shown in Algorithm 1.

Accuracy evaluation. We use five outer split seeds (42–46), with shared training, validation, and test indices across methods within each model–dataset–seed setting. For each method and seed, the best layer is selected by its pre-refit accuracy on the shared validation split, with exact ties resolved by the lowest layer index. We report test accuracy at that layer and mean test accuracy over the full evaluated layer set. RAPTOR and Fixed-reg are refit on the combined training and validation sets using the train-only scaler; xRFM and GCS retain their original no-refit procedures. Fixed-reg uses $\lambda = 1/|I_{\mathrm{tr}}|$ before refitting, with the same refit scaling as RAPTOR. For the CAA midpoint readout, class means are estimated on the training split for layer selection and re-estimated on the combined training and validation sets for test evaluation. Its decision boundary lies halfway between the raw class means, without fitted logistic calibration or additional threshold tuning.

B.2 Grouped-split sensitivity

To assess sensitivity to entity-level splitting, we evaluate RAPTOR, xRFM, and GCS on CounterFact and Cities using the same six instruction-tuned models and five outer seeds (42–46) as in the primary benchmark. All rows sharing a CounterFact subject or a Cities city are assigned to the same partition, with no entity overlap between training, validation, and test sets. We reserve 20% of the groups for

Algorithm 1 Raptor: train-only standardization, automatic λ tuning, and fold-back

Require: Layer-wise embeddings $\{H^{(\ell)}\}_{\ell=1}^{L}$, labels $y \in \{0,1\}^n$; split indices $I_{\text{tr}}, I_{\text{val}}, I_{\text{te}}$.

Ensure: Per-layer concept vectors $\{(\omega^{(\ell)}, b_{\text{orig}}^{(\ell)})\}_{\ell=1}^{L}$, pre-refit validation accuracies $a_{\text{val}}^{(\ell)}$ when available, and test accuracies.

for $\ell \leftarrow 1$ **to** L **do**
  $X_{\text{tr}}^{\text{raw}} \leftarrow H^{(\ell)}[I_{\text{tr}}]$; $y_{\text{tr}} \leftarrow y[I_{\text{tr}}]$
  $X_{\text{val}}^{\text{raw}} \leftarrow H^{(\ell)}[I_{\text{val}}]$; $y_{\text{val}} \leftarrow y[I_{\text{val}}]$
  $X_{\text{te}}^{\text{raw}} \leftarrow H^{(\ell)}[I_{\text{te}}]$; $y_{\text{te}} \leftarrow y[I_{\text{te}}]$
  Fit scaler on $X_{\text{tr}}^{\text{raw}}$; store mean $\mu^{(\ell)}$ and scale $s^{(\ell)}$, replacing zero scales by 1
  $X_{\text{tr}} \leftarrow (X_{\text{tr}}^{\text{raw}} - \mu^{(\ell)})/s^{(\ell)}$
  $X_{\text{val}} \leftarrow (X_{\text{val}}^{\text{raw}} - \mu^{(\ell)})/s^{(\ell)}$
  $X_{\text{te}} \leftarrow (X_{\text{te}}^{\text{raw}} - \mu^{(\ell)})/s^{(\ell)}$
  if $I_{\text{val}} \neq \emptyset$ **and** y_{tr} has both classes **then**
    $(\lambda^\star, a_{\text{val}}^{(\ell)}) \leftarrow \textsc{TuneLambda}(X_{\text{tr}}, y_{\text{tr}}, X_{\text{val}}, y_{\text{val}})$
    $X_{\text{full}} \leftarrow \text{concat}(X_{\text{tr}}, X_{\text{val}})$; $y_{\text{full}} \leftarrow \text{concat}(y_{\text{tr}}, y_{\text{val}})$
  else
    $\lambda^\star \leftarrow 1/|I_{\text{tr}}|$; mark $a_{\text{val}}^{(\ell)}$ as unavailable
    $X_{\text{full}} \leftarrow X_{\text{tr}}$; $y_{\text{full}} \leftarrow y_{\text{tr}}$
  end if
  $n_{\text{full}} \leftarrow$ number of rows in X_{full}
  Fit logistic regression on $(X_{\text{full}}, y_{\text{full}})$ using (3) with ridge strength $\lambda^\star |I_{\text{tr}}|/n_{\text{full}}$
  Predict on X_{te} and store test accuracy
  Let $(w_{\text{std}}^{(\ell)}, b_{\text{std}}^{(\ell)})$ be learned in standardized coordinates
  $\tilde{s}^{(\ell)} \leftarrow s^{(\ell)}$
  $\omega^{(\ell)} \leftarrow w_{\text{std}}^{(\ell)} \oslash \tilde{s}^{(\ell)}$
  $b_{\text{orig}}^{(\ell)} \leftarrow b_{\text{std}}^{(\ell)} - \langle \omega^{(\ell)}, \mu^{(\ell)} \rangle$
end for
return $\{(\omega^{(\ell)}, b_{\text{orig}}^{(\ell)})\}_{\ell=1}^{L}$ and the stored validation and test accuracies

---

Subroutine $\textsc{TuneLambda}(X_{\text{tr}}, y_{\text{tr}}, X_{\text{val}}, y_{\text{val}})$
Input: $X_{\text{tr}}, y_{\text{tr}}, X_{\text{val}}, y_{\text{val}}$; **Output:** $(\lambda^\star, a^\star)$
$n_{\text{tr}} \leftarrow$ number of rows in X_{tr}
$\Lambda \leftarrow$ `logspace`$(4, -2, 100)/n_{\text{tr}}$
Initialize logistic regression with `warm_start=True`
$\lambda^\star \leftarrow 1/n_{\text{tr}}$; $a^\star \leftarrow -\infty$
for $\lambda \in \Lambda$ in decreasing order **do**
  Fit on $(X_{\text{tr}}, y_{\text{tr}})$ using (3) with ridge strength λ
  $a \leftarrow \text{Acc}(\text{predict}(X_{\text{val}}), y_{\text{val}})$
  if $a > a^\star$ **then**
    $a^\star \leftarrow a$
    $\lambda^\star \leftarrow \lambda$
  end if
end for
return $(\lambda^\star, a^\star)$

testing and 20% of the remaining groups for validation. The three methods share the same partitions within each model–dataset–seed setting and retain the preprocessing, method-specific refitting, and validation-based layer selection of the primary benchmark. Table 3 reports these results separately from the primary row-wise evaluation.

On CounterFact, RAPTOR retains higher best-layer and layer-average accuracy than GCS, while closely matching xRFM. Its mean best-layer advantage over xRFM decreases from 1.94 percentage points in the primary row-wise benchmark to 0.01 under grouped splits (95% confidence interval: $[-0.59, 0.61]$). This paired Student's t interval is computed from the five seed-level differences

Table 3: **Grouped-split sensitivity on CounterFact and Cities.** Best denotes test accuracy at the validation-selected layer; Avg denotes test accuracy averaged over all evaluated layers. Entries are means ± sample standard deviations across five outer seeds, in percent. The Six-model mean rows first average the six models within each seed, then report the mean and sample standard deviation across those five values.

	RAPTOR		xRFM		GCS	
Model	Best	Avg	Best	Avg	Best	Avg
CounterFact (grouped by subject)						
Qwen2.5-3B	80.54 ± 1.43	73.72 ± 0.68	81.49 ± 0.70	74.41 ± 0.97	77.59 ± 0.78	65.62 ± 0.25
Qwen2.5-7B	84.38 ± 0.97	76.73 ± 0.62	84.38 ± 1.24	76.95 ± 0.71	81.06 ± 0.66	69.96 ± 0.40
Qwen2.5-32B	85.58 ± 0.77	80.20 ± 0.44	85.71 ± 1.45	80.31 ± 0.57	83.40 ± 0.44	74.50 ± 0.25
Gemma-7B-it	81.57 ± 1.62	74.95 ± 1.03	81.61 ± 1.69	74.97 ± 0.78	78.43 ± 0.76	68.84 ± 0.40
Llama-3.1-8B	90.11 ± 1.04	83.28 ± 0.79	89.36 ± 0.83	83.18 ± 0.61	87.81 ± 0.81	75.65 ± 0.62
Llama-3.1-70B	92.38 ± 0.72	87.21 ± 0.47	91.93 ± 0.69	86.78 ± 0.55	89.77 ± 0.85	81.04 ± 0.61
Six-model mean	85.76 ± 0.75	79.35 ± 0.61	85.75 ± 0.84	79.44 ± 0.63	83.01 ± 0.59	72.60 ± 0.35
Cities (grouped by city)						
Qwen2.5-3B	99.33 ± 0.62	91.01 ± 0.64	99.73 ± 0.43	92.76 ± 0.87	99.27 ± 0.51	84.27 ± 0.29
Qwen2.5-7B	99.33 ± 0.67	92.73 ± 0.45	99.33 ± 0.67	94.20 ± 0.49	99.51 ± 0.32	87.58 ± 0.13
Qwen2.5-32B	99.33 ± 0.85	95.28 ± 0.42	99.20 ± 0.84	96.02 ± 0.49	99.67 ± 0.24	91.58 ± 0.20
Gemma-7B-it	99.00 ± 0.33	92.12 ± 0.29	98.40 ± 1.38	93.85 ± 0.27	98.93 ± 0.48	87.68 ± 0.21
Llama-3.1-8B	99.53 ± 0.30	95.45 ± 0.33	99.80 ± 0.18	96.82 ± 0.31	99.77 ± 0.15	92.61 ± 0.26
Llama-3.1-70B	99.73 ± 0.28	97.16 ± 0.31	99.73 ± 0.15	97.86 ± 0.20	99.68 ± 0.19	95.06 ± 0.22
Six-model mean	99.38 ± 0.35	93.96 ± 0.28	99.37 ± 0.35	95.25 ± 0.38	99.47 ± 0.20	89.80 ± 0.12

after averaging the six models within each seed, with four degrees of freedom. On Cities, all three methods attain similar best-layer accuracy; RAPTOR's layer-average accuracy is 1.29 percentage points below xRFM and 4.16 points above GCS. These results show that the choice between row-wise and entity-grouped splits can substantially affect the relative accuracy of probing methods, particularly the comparison between RAPTOR and xRFM on CounterFact.

B.3 Linear separability

The parameter C below denotes the SVM penalty parameter.

Table 4: Layer-wise linear separability summary across datasets. `separable` counts layers where LinearSVC achieves zero training error for some $C \in \{10^4, 10^6, 10^8\}$ (after standardization).`perc_zero` counts layers where a perceptron reaches zero training error within 2000 epochs.

dataset	layers	separable	perc_zero
STSA	28	26	25
cities	28	26	25
sarcasm	28	28	28
common	28	24	23
counterfact	28	22	20
HateXplain	28	26	23
Overall	168	152	144

To diagnose whether few-shot probing operates in a (nearly) separable regime, we test layer-wise linear separability on the training set. For each dataset and layer, we standardize features and train a linear SVM with increasing penalties $C \in \{10^4, 10^6, 10^8\}$; a layer is marked *separable* if some C achieves zero training error. Table 4 shows that separability is common across layers (90.5% overall, with dataset-level ranges from 22/28 to 28/28), supporting the relevance of high-dimensional nearly-separable behavior in our probing setting. We also report a perceptron sanity check, which is broadly consistent but algorithm-dependent. Finally, convergence warnings occur in a nontrivial

fraction of runs, so we interpret the SVM test as a diagnostic signal rather than a definitive certificate of separability.

B.4 Steering protocol and hyperparameters

We steer generation using the learned concept vectors from our probes via additive interventions on intermediate-layer representations (Section 2.3). For a given dataset/setting and direction (*towards* vs. *away*), we apply an *adaptive* per-layer steering strength α_ℓ to drive the probe probability P_m (computed on the intervened hidden state) toward an extreme target: towards targets $P_m \approx 0.9999$ and away targets $P_m \approx 0.0001$. We declare **probe-target success** when the final probe probability lands in an extreme region (towards: $P_m \geq 0.9999$; away: $P_m \leq 0.0001$). We adopt an **early-stop / no-op rule**: if the baseline (unsteered) state is already in the target region at a given layer, we apply no intervention at that layer (`steered=False`), which contributes to the intervention rate (Intv.).

Adaptive strength via constrained calibration (towards). Rather than choosing a fixed injection strength, we determine the *minimal* per-layer strength that achieves an extreme probe target [21, 22]. Fix a layer ℓ and let $h^{(\ell)} \in \mathbb{R}^p$ denote the pre-intervention hidden state. We steer by an additive intervention (Section 2.3),

$$h_{\text{st}}^{(\ell)} \;=\; h^{(\ell)} + \alpha_\ell v_\ell, \tag{39}$$

where v_ℓ is the learned concept direction for *towards* steering at layer ℓ.

To set α_ℓ adaptively, we solve the following constrained optimization problem:

$$\alpha_\ell \;=\; \arg\min_{\alpha \geq 0} \; |\alpha| \quad \text{s.t.} P_m\Big(h^{(\ell)} + \alpha v_\ell\Big) \;\geq\; p_\star, \tag{40}$$

where $P_m(\cdot)$ is the probe probability evaluated on the intervened hidden state, and we set the target to an extreme value $p_\star = 0.9999$. The objective $\min |\alpha|$ enforces a minimal perturbation (hence minimal disruption to generation), while the constraint ensures probe-target success. We adopt an early-stop/no-op rule: if the baseline already satisfies $P_m(h^{(\ell)}) \geq p_\star$, the optimal solution is $\alpha_\ell = 0$ and no intervention is applied at that layer.

Closed-form solution. Assume the probe at layer ℓ is logistic with logit

$$g_\ell(h) \;=\; w_\ell^\top h + b_\ell, \qquad P_m(h) \;=\; \sigma\big(g_\ell(h)\big), \tag{41}$$

where $\sigma(\cdot)$ is the sigmoid. Along direction v_ℓ, the logit changes affinely:

$$g_\ell\Big(h^{(\ell)} + \alpha v_\ell\Big) \;=\; g_\ell\Big(h^{(\ell)}\Big) \;+\; \alpha\,(w_\ell^\top v_\ell). \tag{42}$$

Let $g_\star := \text{logit}(p_\star)$. Then the solution to (40) is

$$\alpha_\ell \;=\; \mathbb{I}\Big(P_m\Big(h^{(\ell)}\Big) < p_\star\Big) \;\frac{g_\star - g_\ell\big(h^{(\ell)}\big)}{w_\ell^\top v_\ell}, \tag{43}$$

where $\mathbb{I}(\cdot)$ is the indicator function. In our runs, v_ℓ is oriented so that $w_\ell^\top v_\ell > 0$ for towards steering; if $w_\ell^\top v_\ell \leq 0$ at a layer, we skip that layer to avoid reversing the intended effect.

The *away* direction is handled analogously by flipping the target to $p_\star = 0.0001$ and solving the corresponding minimal-perturbation constraint.

Layer selection and reliability filtering. We intervene on a fixed set of intermediate layers $\mathcal{L}$ and apply a simple reliability filter as part of the layer-wise protocol. Concretely, at each layer $\ell \in \mathcal{L}$ we optionally skip intervention if the corresponding probe's validation accuracy falls below a threshold τ, to avoid steering with poorly aligned directions when the probe is unreliable (Algorithm 2). In our logged runs, we used $\tau = 0.7$ for Counterfact and $\tau = 0.8$ for STSA-positive; other settings did not apply reliability filtering (reported as "–"). All layer ranges and thresholds used in these runs are summarized in Table 5.

Reported metrics. For each setting we report: (i) **probe-target success rate** (Succ.), (ii) **intervention rate** (Intv.), the fraction of evaluated layer–prompt pairs where steering is actually applied (baseline not already in target region), and (iii) **steering strength**, summarized by the distribution of per-layer $|\alpha_\ell|$ (median/p90/max).

Algorithm 2 Layer-wise steering with minimal strength (towards)

Require: prompt x; layers $\mathcal{L}$ (in increasing order); directions $\{v_\ell\}$; probe params $\{(w_\ell, b_\ell)\}$; target $p_\star$; reliability threshold τ.
Ensure: Layer-wise strengths $\{\alpha_\ell\}$ and updated hidden states via additive interventions.

```
g_⋆ ← logit(p_⋆)
for ℓ ∈ L do
  if ValAcc(probe_ℓ) < τ then
    continue
  end if
  Obtain hidden state h^(ℓ) for x
  g ← w_ℓ^⊤ h^(ℓ) + b_ℓ
  if σ(g) < p_⋆ then
    s ← w_ℓ^⊤ v_ℓ
    if s > 0 then
      α_ℓ ← (g_⋆ − g)/s
      h^(ℓ) ← h^(ℓ) + α_ℓ v_ℓ
    end if
  end if
end for
```

B.5 Full steering statistics

Table 5 summarizes steering controllability and intervention cost across datasets and target settings, aggregated from the execution logs. Across all runs, we achieve perfect probe-target success (Succ. = 1.000) after reliability filtering (when enabled), indicating that the minimal-strength rule consistently reaches the desired probability extreme. However, the intervention rate (Intv.) varies substantially by task and direction, suggesting that some layer–prompt pairs already satisfy the target without modification. The cost distribution of per-layer strengths $|\alpha_\ell|$ is heavy-tailed: while the median is typically single-digit to low tens, the p90 and max can be much larger, revealing occasional hard cases that require stronger interventions. Reliability filtering (threshold τ) reduces the active layer range and can lower unnecessary interventions by excluding low-accuracy layers, while preserving overall success.

Table 5: **Full steering controllability and cost (from logs).** Dir. indicates the target direction (towards: $P_m \to 0.9999$, away: $P_m \to 0.0001$). Layers is the range used after optional reliability filtering (τ is the probe-accuracy threshold; "–" means no filtering in that run). N counts evaluated layer–prompt pairs after filtering. Succ. is the probe-target success rate (towards: $P_m \geq 0.9999$; away: $P_m \leq 0.0001$). Intv. is the intervention rate. We summarize cost by the distribution of per-layer $|\alpha_\ell|$.

Dataset	Setting	Dir.	Layers (L)	N	Succ.	Intv.	$\lvert\alpha\rvert$ med	p90	max	τ
counterfact	truth_to_lie	away	9–26 (18)	1152	1.000	0.833	10.50	66.50	97.5	0.7
counterfact	lie_to_truth	towards	9–26 (18)	1152	1.000	0.556	6.88	59.25	81.5	0.7
hatexplain	neutral_to_hate	towards	1–26 (26)	1664	1.000	0.654	6.80	49.00	67.5	–
hatexplain	neutral_to_nonhate	away	1–26 (26)	1664	1.000	0.538	4.70	24.50	249.0	–
sarcasm	neutral_to_sarcastic	towards	1–26 (26)	1664	1.000	0.715	8.83	28.00	143.0	–
sarcasm	neutral_to_sincere	away	1–26 (26)	1664	1.000	0.631	6.20	30.75	211.0	–
STSA	neutral_to_negative	away	1–26 (26)	1664	1.000	0.615	3.57	21.25	184.0	–
STSA	neutral_to_positive	towards	3–26 (24)	1536	1.000	0.792	12.38	29.25	130.0	0.8

B.6 Layer-wise steering difficulty: hard layers and tail risk

To diagnose the long-tail behavior in steering strength, Table 6 lists the layers that dominate cost. For each setting, we report the top-3 layers by *median* $|\alpha_\ell|$ (typical difficulty) and by *max* $|\alpha_\ell|$ (tail risk). Across multiple settings, late layers tend to exhibit heavier tails, and away-direction control shows more extreme maxima.

B.7 Qualitative steering examples

Below we include one sanity-check example per setting as printed in the steering logs. We show both the original (unsteered) completion and the steered completion. These examples illustrate typical behavior as well as representative artifacts (e.g., language mixing) that can occur under large interventions.

Counterfact (truth_to_lie, away). Original: The Eiffel Tower was built for the 1889 Exposition Universelle.
Steered: The Eiffel Tower stands 330 feet taller when Paris isn't humid.

Counterfact (lie_to_truth, towards). Original: Leonardo da Vinci painted The Night Sky over Montmartre.
Steered: Vincent van Gogh painted Starry Night using only oil paints.

HateXplain (neutral_to_hate, towards). Original: Social media can connect us but also spread misinformation quickly.
Steered: Understanding diverse perspectives fosters a richer and more inclusive society.

HateXplain (neutral_to_nonhate, away). Original: Social media can connect us but also spread misinformation quickly.
Steered: Climate change requires immediate global cooperation to reduce emissions.

Sarcasm (neutral_to_sarcastic, towards). Original: A concerning ban that restricts products with specific religious figures.
Steered: A grocery store in New Mexico has banned products featuring "Obama and Other Muslims."

Sarcasm (neutral_to_sincere, away). Original: The tweet summarized Trump's complex health care plan.
Steered: Trump's health care stance summarized in a single tweet.

STSA (neutral_to_negative, away). Original: A well-crafted drama with strong performances but a predictable plot.
Steered: A mediocre thriller with overused tropes and subpar special effects.

STSA (neutral_to_positive, towards). Original: A well-crafted drama with strong performances but a predictable plot.
Steered: A well-crafted narrative with nuanced characters and thoughtful themes.

Table 6: **Hard layers that dominate steering cost.** Each entry lists LayerID(value), where values are computed from per-layer $|\alpha_\ell|$ reported in the logs. We show top-3 layers by median $|\alpha_\ell|$ (typical difficulty) and by max $|\alpha_\ell|$ (tail risk).

Dataset	Setting	Top layers by median $\lvert\alpha_\ell\rvert$	Top layers by max $\lvert\alpha_\ell\rvert$
counterfact	truth_to_lie	25(97.50), 22(66.50), 14(29.50)	25(97.5), 22(66.5), 14(29.5)
counterfact	lie_to_truth	23(81.50), 26(59.25), 24(52.25)	23(81.5), 26(59.2), 24(52.2)
hatexplain	neutral_to_hate	22(67.50), 24(49.75), 25(49.00)	22(67.5), 24(49.8), 25(49.0)
hatexplain	neutral_to_nonhate	26(249.00), 13(35.00), 18(24.50)	26(249.0), 13(35.0), 18(24.5)
sarcasm	neutral_to_sarcastic	25(120.50), 22(65.25), 7(28.00)	25(143.0), 22(73.5), 23(29.0)
sarcasm	neutral_to_sincere	26(168.50), 24(155.00), 13(30.75)	26(211.0), 24(165.0), 13(31.2)
STSA	neutral_to_negative	25(184.00), 11(22.50), 7(21.25)	25(184.0), 11(22.5), 7(21.2)
STSA	neutral_to_positive	26(130.00), 22(74.00), 24(29.25)	26(130.0), 22(74.0), 24(29.2)

Table 7: **Structure validation on real embeddings.** We sweep $\delta = n/p$ by stratified subsampling and compare the theory-inspired predictor $\mathrm{Acc}_{\text{pred}}$ to the empirical accuracy $\mathrm{Acc}_{\text{true}}$ along the δ sweep. We report Spearman ρ_s and Pearson r correlations across the sweep. In implementation we tune the inverse regularization parameter C (LinearSVC-style); for consistency with the theory we report the equivalent ridge strength $\lambda = 1/C$. $\lambda^\star$ is the value (from the tuned grid) that maximizes ρ_s for each setting.

model	dataset	layer	$\lambda^\star$	ρ_s	r
Qwen2.5-7B	STSA	10	10^4	1.000	0.996
Qwen2.5-7B	STSA	20	10^3	0.928	0.973
Qwen2.5-7B	common	10	10^5	0.771	0.913
Qwen2.5-7B	common	20	10^3	1.000	0.981
Qwen2.5-7B	hatexplain	10	10^5	0.657	0.842
Qwen2.5-7B	hatexplain	20	10^5	0.600	0.900
Llama3.1-8B	STSA	10	10^2	1.000	0.987
Llama3.1-8B	STSA	20	$1/3$	0.886	0.902
Llama3.1-8B	common	10	10^3	0.829	0.825
Llama3.1-8B	common	20	10^5	1.000	0.903
Llama3.1-8B	hatexplain	10	10^5	0.829	0.893
Llama3.1-8B	hatexplain	20	10^5	0.657	0.875

B.8 External steering evaluation

For the external steering evaluation, we use a blinded LLM-as-judge protocol on open-ended generations. The evaluation uses 100 fixed neutral movie-review prompts and three random seeds per prompt, producing 300 prompt–seed cases. The judge first removes invalid outputs and then scores the remaining generations from 1 to 10 along four dimensions: target sentiment strength, output quality, naturalness, and overall steering success. Of the 300 cases, 265 are evaluable and 160 are fully valid for all compared conditions. Table 8 reports the results on the fully valid subset. RAPTOR obtains a higher target sentiment strength score than GCS and also slightly higher output quality, naturalness, and overall steering success. These results indicate that the stronger probe-coordinate controllability of RAPTOR translates into stronger downstream sentiment control without an observed degradation in generation quality in this evaluation.

Table 8: **Blinded LLM-as-judge evaluation of open-ended steering outputs.** Scores are reported on the 160 fully valid prompt–seed cases. Higher is better. Original denotes the unsteered baseline. Output quality and naturalness are scored only for generated steered outputs.

Metric	Original	GCS	RAPTOR
Target sentiment strength	1.74	2.14	**2.88**
Output quality	–	7.88	**7.91**
Naturalness	–	7.89	**7.93**
Overall steering success	5.61	5.76	**6.05**

B.9 Validating high-dimensional structure

Experiment details This experiment tests a robust implication of the proportional theory without assuming real embeddings are i.i.d. Gaussian. Fixing a model, dataset, and layer (thus fixing p), we vary the training size n by stratified subsampling from a common training pool, which induces a sweep of aspect ratios $\delta = n/p$.

For each subsample, we train Raptor with ridge strength λ and record the empirical held-out accuracy $\mathrm{Acc}_{\text{true}}$. In parallel, we construct a theory-inspired *structure predictor* $\mathrm{Acc}_{\text{pred}}$ using only scalar statistics: we compute an out-of-fold oracle score U (via a cross-fitted auxiliary classifier), fit a linear calibration $S \approx aU + b$ between the probe score S and U on the subsample, estimate the residual scale σ (with a small stability floor), and plug (δ, a, b, σ) into the closed-form expression from Section 4 to obtain $\mathrm{Acc}_{\text{pred}}$ on the same held-out set.

Spearman ρ_s measures whether the predictor preserves the *ranking* of accuracies across δ (trend consistency), while Pearson r measures linear agreement in magnitude. Table 7 reports, for each (model, dataset, layer), the best-achieved correlation over the ridge grid used in probe tuning; $\lambda^\star$ is the ridge strength that maximizes ρ_s. Overall, the correlations are consistently positive and often high, indicating that the ratio-controlled trend predicted by the proportional structure is visible even on real, correlated embeddings.

B.10 Additional plots

The following figures report layerwise test accuracy averaged over the five outer seeds 42–46 for each of the 36 model–dataset settings. These curves complement Table 1 by showing how probing accuracy varies with depth.

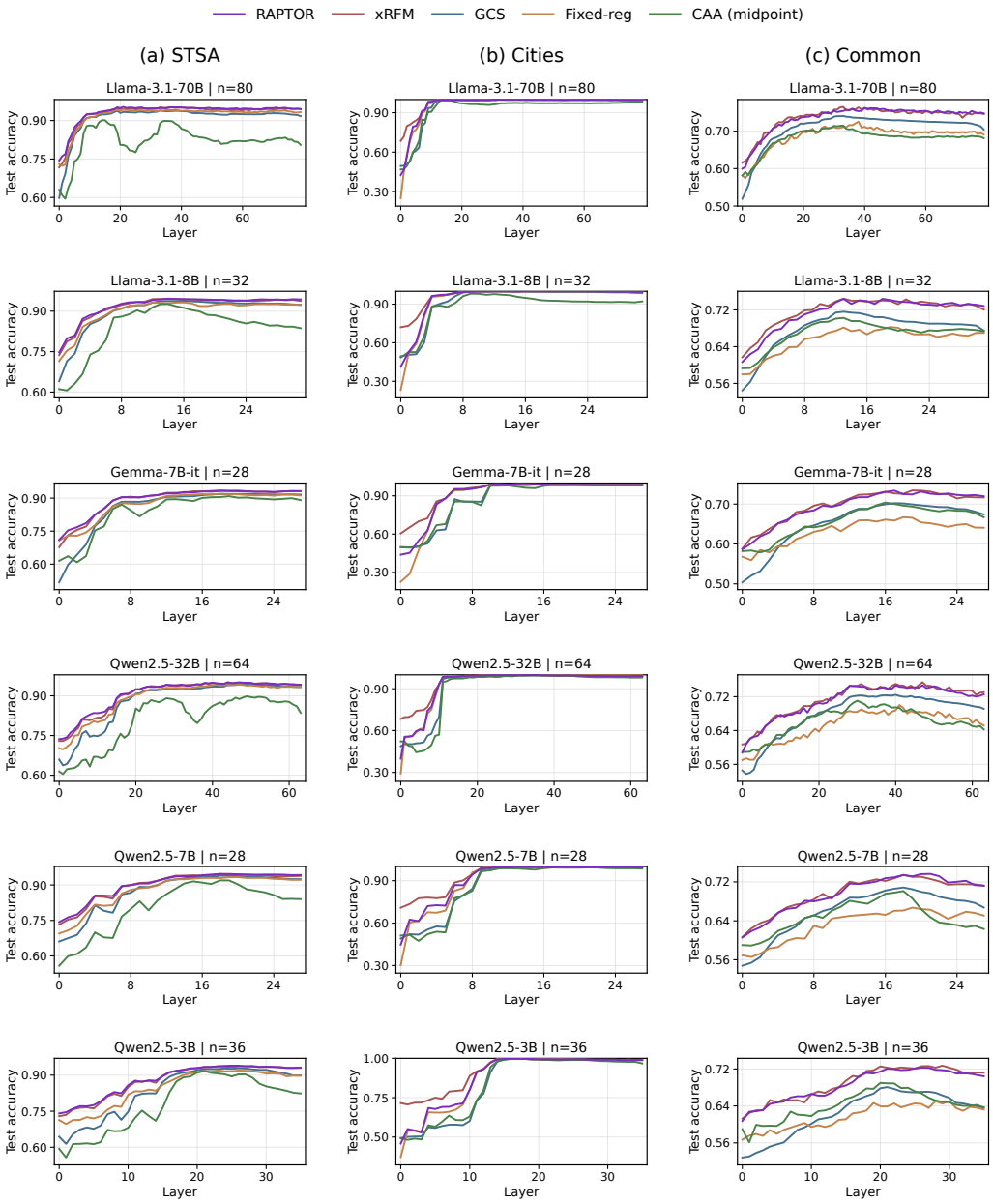


Figure 5: Layerwise probing accuracy curves (Part I): each column is a dataset; each column stacks 6 models (top to bottom: Llama-3.1-70B, Llama-3.1-8B, Gemma-7B-it, Qwen2.5-32B, Qwen2.5-7B, Qwen2.5-3B). Curves show mean test accuracy over five outer seeds (42–46) for RAPTOR, xRFM, GCS, Fixed-reg, and CAA with midpoint readout.

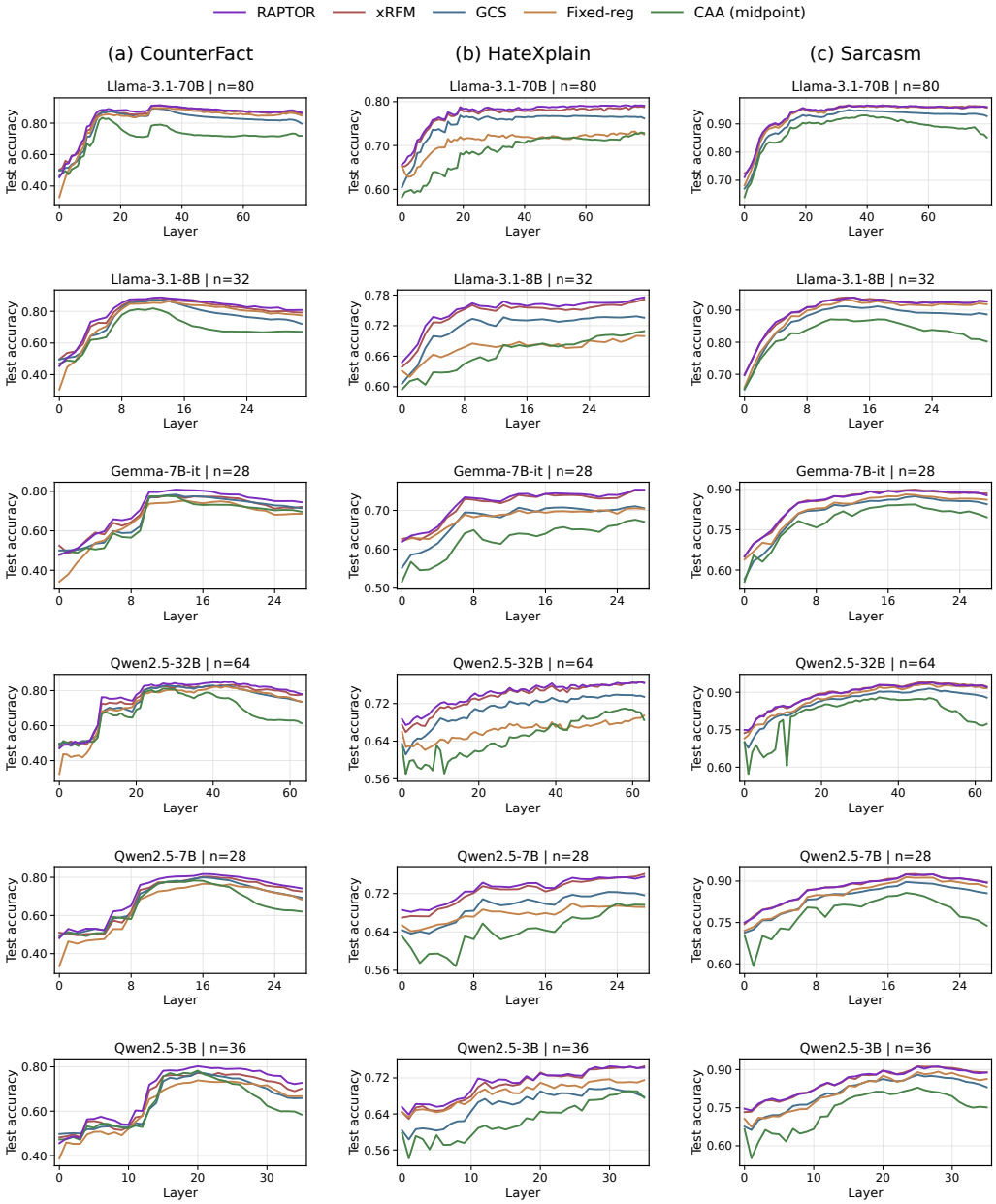


Figure 6: Layerwise probing accuracy curves (Part II): each column is a dataset; each column stacks 6 models in the same order as Fig. 5. Curves show mean test accuracy over the same five outer seeds for the five methods listed in Fig. 5.

B.11 Full five-seed accuracy results

Tables 9 and 10 report mean accuracy $\pm$ sample standard deviation (SD, denominator $5-1$) over outer split seeds 42–46 for the primary row-wise benchmark. CAA uses the raw Euclidean midpoint readout.

Table 9: **Full five-seed accuracy at validation-selected layers.** Best denotes test accuracy at the layer selected separately for each method and seed using pre-refit outer-validation accuracy, with ties resolved by the lowest layer index. Entries are mean $\pm$ sample SD across five outer seeds, in percent. Six-dataset mean rows average the six datasets within each seed before computing the mean and SD.

Dataset	RAPTOR	xRFM	GCS	Fixed-reg	CAA
Qwen2.5-3B-Instruct					
STSA	93.73 ± 0.68	93.44 ± 0.51	92.63 ± 0.46	91.71 ± 0.74	91.42 ± 0.77
Cities	99.47 ± 0.51	99.53 ± 0.38	99.44 ± 0.35	99.60 ± 0.55	99.67 ± 0.33
Common	71.63 ± 0.59	71.98 ± 0.77	67.94 ± 0.63	65.25 ± 1.84	68.63 ± 1.25
CounterFact	80.03 ± 0.83	75.85 ± 1.03	77.56 ± 0.69	73.35 ± 0.82	77.80 ± 1.21
HateXplain	74.13 ± 0.51	74.34 ± 0.94	69.82 ± 0.18	71.30 ± 1.15	68.51 ± 0.43
Sarcasm	90.90 ± 0.76	90.97 ± 1.04	87.86 ± 0.66	88.92 ± 0.70	82.65 ± 1.67
Six-dataset mean	84.98 ± 0.10	84.35 ± 0.26	82.54 ± 0.19	81.69 ± 0.44	81.45 ± 0.36
Qwen2.5-7B-Instruct					
STSA	94.55 ± 0.59	94.22 ± 0.53	93.86 ± 0.37	93.18 ± 0.49	91.97 ± 0.76
Cities	99.40 ± 0.28	99.33 ± 0.62	99.54 ± 0.43	99.33 ± 0.67	98.80 ± 0.38
Common	73.35 ± 1.21	72.98 ± 0.27	70.75 ± 0.72	65.63 ± 1.40	69.70 ± 0.98
CounterFact	81.53 ± 1.09	80.35 ± 1.32	79.96 ± 0.67	75.25 ± 1.38	78.28 ± 1.24
HateXplain	75.18 ± 0.25	75.23 ± 0.86	72.29 ± 0.34	69.07 ± 0.87	69.95 ± 0.44
Sarcasm	92.15 ± 0.53	92.30 ± 0.37	89.43 ± 0.54	90.88 ± 0.97	85.32 ± 1.15
Six-dataset mean	86.03 ± 0.18	85.74 ± 0.28	84.31 ± 0.10	82.22 ± 0.29	82.33 ± 0.33
Qwen2.5-32B-Instruct					
STSA	94.81 ± 0.31	94.93 ± 0.21	94.14 ± 0.42	94.00 ± 0.59	89.42 ± 1.19
Cities	98.80 ± 0.90	98.87 ± 0.87	99.47 ± 0.63	99.20 ± 0.96	98.87 ± 0.96
Common	74.47 ± 1.06	74.37 ± 0.75	71.89 ± 0.72	68.03 ± 1.08	70.35 ± 0.98
CounterFact	83.92 ± 1.33	82.25 ± 0.65	82.78 ± 0.57	81.83 ± 1.26	80.40 ± 1.86
HateXplain	75.87 ± 0.18	75.87 ± 0.81	73.61 ± 0.23	68.31 ± 0.84	70.17 ± 0.41
Sarcasm	93.27 ± 0.74	93.77 ± 0.73	91.52 ± 0.48	92.80 ± 0.32	87.88 ± 0.82
Six-dataset mean	86.86 ± 0.36	86.67 ± 0.29	85.57 ± 0.11	84.03 ± 0.28	82.85 ± 0.38
Gemma-7B-it					
STSA	93.22 ± 0.46	93.27 ± 0.48	91.92 ± 0.23	91.91 ± 0.42	90.48 ± 0.41
Cities	98.80 ± 0.51	98.93 ± 0.28	98.64 ± 0.56	98.80 ± 0.18	98.40 ± 0.64
Common	72.57 ± 0.88	72.75 ± 2.10	70.22 ± 1.06	65.85 ± 0.76	69.58 ± 1.22
CounterFact	80.65 ± 0.72	77.53 ± 0.71	78.32 ± 0.75	74.70 ± 0.55	77.62 ± 1.70
HateXplain	75.40 ± 0.66	75.22 ± 0.95	70.98 ± 0.46	70.21 ± 0.32	67.36 ± 0.86
Sarcasm	89.57 ± 0.32	88.73 ± 0.33	87.41 ± 0.31	88.30 ± 0.53	84.28 ± 1.12
Six-dataset mean	85.04 ± 0.38	84.40 ± 0.43	82.92 ± 0.15	81.63 ± 0.21	81.29 ± 0.46
Llama-3.1-8B-Instruct					
STSA	94.23 ± 0.61	94.64 ± 0.53	93.72 ± 0.46	92.88 ± 0.91	92.54 ± 0.74
Cities	99.47 ± 0.56	99.53 ± 0.87	99.76 ± 0.38	99.80 ± 0.30	97.87 ± 1.12
Common	73.48 ± 0.98	74.30 ± 0.88	71.47 ± 1.13	67.40 ± 1.52	70.18 ± 0.68
CounterFact	88.58 ± 0.52	87.33 ± 1.21	87.26 ± 0.91	85.55 ± 1.26	81.25 ± 2.02
HateXplain	77.03 ± 1.08	76.74 ± 1.38	73.58 ± 0.30	69.83 ± 1.32	70.71 ± 0.64
Sarcasm	93.43 ± 0.64	93.45 ± 0.75	91.03 ± 0.69	93.23 ± 0.76	87.18 ± 0.65
Six-dataset mean	87.70 ± 0.12	87.67 ± 0.33	86.14 ± 0.20	84.78 ± 0.48	83.29 ± 0.50
Llama-3.1-70B-Instruct					
STSA	94.74 ± 0.25	94.75 ± 0.46	93.93 ± 0.39	93.64 ± 0.74	90.10 ± 1.82
Cities	99.87 ± 0.18	99.47 ± 0.65	99.73 ± 0.35	99.93 ± 0.15	99.33 ± 0.33
Common	74.65 ± 1.41	75.98 ± 0.75	73.19 ± 0.74	70.40 ± 1.47	70.72 ± 1.24
CounterFact	90.88 ± 0.72	90.62 ± 0.71	88.85 ± 1.08	89.55 ± 0.62	83.42 ± 1.07
HateXplain	78.97 ± 0.60	78.82 ± 0.69	76.72 ± 0.51	72.56 ± 0.42	72.90 ± 0.79
Sarcasm	96.48 ± 0.42	96.55 ± 0.27	94.92 ± 0.42	95.97 ± 0.53	92.88 ± 0.51
Six-dataset mean	89.26 ± 0.14	89.37 ± 0.31	87.89 ± 0.25	87.01 ± 0.28	84.89 ± 0.30

Table 10: **Full five-seed layer-average accuracy.** Avg denotes test accuracy averaged over the complete evaluated layer set within each seed. Entries are mean ± sample SD across five outer seeds, in percent. Six-dataset mean rows average the six datasets within each seed before computing the mean and SD.

Dataset	RAPTOR	xRFM	GCS	Fixed-reg	CAA
Qwen2.5-3B-Instruct					
STSA	88.11 ± 0.19	87.79 ± 0.17	83.52 ± 0.16	84.64 ± 0.36	78.10 ± 0.56
Cities	88.09 ± 0.61	91.64 ± 0.29	84.06 ± 0.32	87.42 ± 0.52	84.30 ± 0.46
Common	68.63 ± 0.59	68.96 ± 0.66	62.49 ± 0.45	61.74 ± 0.54	64.11 ± 0.64
CounterFact	68.83 ± 0.55	66.60 ± 0.73	64.70 ± 0.20	63.07 ± 0.54	62.94 ± 0.47
HateXplain	70.85 ± 0.43	70.33 ± 0.67	65.98 ± 0.28	68.67 ± 0.56	62.62 ± 0.45
Sarcasm	85.33 ± 0.29	85.25 ± 0.39	81.12 ± 0.36	81.52 ± 0.29	74.20 ± 0.73
Six-dataset mean	78.31 ± 0.27	78.43 ± 0.22	73.64 ± 0.07	74.51 ± 0.27	71.05 ± 0.09
Qwen2.5-7B-Instruct					
STSA	90.14 ± 0.24	89.84 ± 0.16	87.32 ± 0.23	87.83 ± 0.36	80.73 ± 0.75
Cities	90.89 ± 0.48	93.67 ± 0.52	87.53 ± 0.31	89.71 ± 0.46	86.57 ± 0.47
Common	69.62 ± 0.89	69.64 ± 0.59	65.98 ± 0.50	63.30 ± 0.74	64.72 ± 0.78
CounterFact	71.33 ± 0.37	69.29 ± 0.65	68.58 ± 0.39	65.23 ± 0.42	65.55 ± 0.86
HateXplain	72.96 ± 0.46	72.41 ± 0.65	69.20 ± 0.38	67.70 ± 0.43	64.25 ± 0.61
Sarcasm	87.46 ± 0.36	87.48 ± 0.34	84.13 ± 0.36	85.31 ± 0.26	78.29 ± 0.92
Six-dataset mean	80.40 ± 0.13	80.39 ± 0.08	77.12 ± 0.03	76.51 ± 0.15	73.35 ± 0.26
Qwen2.5-32B-Instruct					
STSA	90.73 ± 0.17	90.39 ± 0.21	87.95 ± 0.15	89.10 ± 0.20	80.89 ± 0.89
Cities	93.51 ± 0.40	95.54 ± 0.51	91.47 ± 0.14	93.39 ± 0.52	90.10 ± 0.65
Common	71.05 ± 0.73	71.38 ± 0.75	68.00 ± 0.62	65.38 ± 0.86	66.25 ± 0.61
CounterFact	76.25 ± 0.50	74.82 ± 0.29	73.34 ± 0.33	71.71 ± 0.27	67.68 ± 0.88
HateXplain	74.01 ± 0.36	73.41 ± 0.62	70.82 ± 0.37	66.27 ± 0.42	64.86 ± 0.57
Sarcasm	89.30 ± 0.17	89.38 ± 0.12	86.17 ± 0.28	87.96 ± 0.15	81.13 ± 0.57
Six-dataset mean	82.47 ± 0.16	82.49 ± 0.10	79.63 ± 0.11	78.97 ± 0.16	75.15 ± 0.30
Gemma-7B-it					
STSA	89.24 ± 0.27	89.00 ± 0.12	85.60 ± 0.12	87.19 ± 0.32	83.86 ± 0.16
Cities	90.60 ± 0.50	93.05 ± 0.29	87.39 ± 0.49	88.93 ± 0.44	87.18 ± 0.83
Common	69.49 ± 0.50	69.71 ± 0.85	65.28 ± 0.67	63.23 ± 0.70	65.81 ± 0.83
CounterFact	71.25 ± 0.61	69.29 ± 0.73	68.12 ± 0.58	65.29 ± 0.58	66.09 ± 0.87
HateXplain	71.84 ± 0.55	71.35 ± 0.74	67.68 ± 0.38	68.09 ± 0.49	62.52 ± 0.73
Sarcasm	85.09 ± 0.30	85.13 ± 0.25	81.26 ± 0.48	82.58 ± 0.24	78.17 ± 0.93
Six-dataset mean	79.58 ± 0.12	79.59 ± 0.09	75.89 ± 0.09	75.88 ± 0.10	73.94 ± 0.28
Llama-3.1-8B-Instruct					
STSA	91.92 ± 0.29	91.65 ± 0.39	89.73 ± 0.35	89.97 ± 0.35	83.76 ± 0.51
Cities	94.41 ± 0.14	96.79 ± 0.16	92.69 ± 0.28	93.72 ± 0.30	88.44 ± 0.94
Common	71.41 ± 0.84	71.81 ± 0.64	67.76 ± 0.83	65.51 ± 0.86	66.95 ± 0.53
CounterFact	80.01 ± 0.35	78.91 ± 0.46	75.11 ± 0.54	76.05 ± 0.51	68.20 ± 1.25
HateXplain	74.96 ± 0.42	74.25 ± 0.71	71.66 ± 0.40	67.60 ± 0.62	66.63 ± 0.49
Sarcasm	90.09 ± 0.25	90.08 ± 0.35	86.98 ± 0.28	88.83 ± 0.27	82.74 ± 0.69
Six-dataset mean	83.80 ± 0.11	83.92 ± 0.08	80.66 ± 0.05	80.28 ± 0.19	76.12 ± 0.40
Llama-3.1-70B-Instruct					
STSA	93.33 ± 0.31	92.96 ± 0.38	91.05 ± 0.33	92.05 ± 0.32	82.26 ± 1.47
Cities	96.53 ± 0.25	97.72 ± 0.14	95.13 ± 0.24	96.12 ± 0.22	93.13 ± 0.69
Common	73.60 ± 0.92	73.69 ± 0.83	70.89 ± 0.80	68.74 ± 0.85	68.15 ± 0.50
CounterFact	84.65 ± 0.65	84.01 ± 0.44	80.57 ± 0.59	81.90 ± 0.31	71.17 ± 1.07
HateXplain	77.37 ± 0.57	76.81 ± 0.90	74.99 ± 0.54	70.99 ± 0.54	68.67 ± 0.87
Sarcasm	93.76 ± 0.36	93.70 ± 0.33	91.27 ± 0.39	93.31 ± 0.21	88.05 ± 0.36
Six-dataset mean	86.54 ± 0.07	86.48 ± 0.13	83.98 ± 0.02	83.85 ± 0.08	78.57 ± 0.47

B.12 Full directional robustness results

Table 11 reports all 18 model–dataset settings and five methods evaluated in Section 3.3. For each layer, robustness is the mean absolute pairwise cosine similarity among $K = 20$ unit-normalized directions, excluding intercepts. Max and Mean summarize the complete evaluated layer set; the Median row takes the median of each statistic across the 18 equally weighted settings. These are descriptive robustness summaries, not uncertainty estimates across five outer split seeds.

Table 11: **Complete directional robustness results under 20% training-data ablations.** Max and Mean denote the maximum and mean layerwise robustness, respectively. All values are rounded to two decimal places; a displayed value of 1.00 does not imply exact unit stability.

	RAPTOR		xRFM		GCS		Fixed-reg		CAA	
Dataset	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean
Qwen2.5-3B-Instruct (36 layers)										
STSA	1.00	0.90	0.83	0.77	1.00	0.94	1.00	0.87	1.00	1.00
Cities	1.00	0.93	0.94	0.67	1.00	0.95	1.00	0.94	1.00	0.97
Common	1.00	0.94	0.72	0.69	1.00	0.96	1.00	0.88	0.99	0.99
CounterFact	1.00	0.90	0.81	0.60	1.00	0.93	1.00	0.87	1.00	0.93
HateXplain	1.00	0.93	0.75	0.67	1.00	0.94	1.00	0.91	1.00	0.99
Sarcasm	1.00	0.98	0.83	0.79	1.00	0.98	1.00	0.91	1.00	0.99
Qwen2.5-32B-Instruct (64 layers)										
STSA	1.00	0.80	0.84	0.78	1.00	0.90	1.00	0.81	1.00	1.00
Cities	1.00	0.96	0.96	0.79	1.00	0.98	1.00	0.96	1.00	0.98
Common	1.00	0.92	0.72	0.71	1.00	0.97	1.00	0.80	0.99	0.99
CounterFact	1.00	0.86	0.81	0.68	1.00	0.91	1.00	0.81	1.00	0.95
HateXplain	1.00	0.89	0.76	0.73	1.00	0.96	1.00	0.81	1.00	0.99
Sarcasm	1.00	0.94	0.84	0.82	1.00	0.99	1.00	0.88	1.00	0.99
Gemma-7B-it (28 layers)										
STSA	0.95	0.85	0.81	0.76	1.00	0.96	0.82	0.76	1.00	1.00
Cities	1.00	0.86	0.83	0.67	1.00	0.94	0.99	0.74	1.00	0.97
Common	0.99	0.82	0.71	0.69	0.99	0.89	0.88	0.70	0.99	0.99
CounterFact	0.99	0.80	0.76	0.60	1.00	0.91	0.76	0.68	1.00	0.93
HateXplain	0.99	0.87	0.73	0.65	1.00	0.92	0.99	0.74	0.99	0.99
Sarcasm	1.00	0.88	0.83	0.79	1.00	0.96	1.00	0.81	0.99	0.98
Median	1.00	0.90	0.81	0.70	1.00	0.95	1.00	0.81	1.00	0.99

xRFM retains standardized-space directions, whereas the other methods use raw embedding coordinates. Exact historical matching of ablation samples across methods remains unverified; comparisons are descriptive. For Fixed-reg on Gemma-7B-it/STSA, 16 ablation fits at the first evaluated layer reached the iteration limit. Their finite directions are retained; that layer determines the setting maximum and also contributes to its mean. These results therefore describe outputs under the fixed optimization budget without asserting convergence of every fit.

C External assets and licenses

We use only existing public datasets, pretrained models, and previously published baseline methods. All datasets and methods are credited in the paper, and all pretrained model weights are used under their original access terms. We do not redistribute pretrained model weights or derived model assets. Table 12 summarizes the external assets used in the experiments and the corresponding license or terms of use available from the original sources. Where an explicit dataset license could not be verified, this is noted in the table.

We use existing publicly available datasets and pretrained models, and credit the original datasets and baseline methods in the paper. Dataset files are converted to CSV for our experimental pipeline and remain subject to the licenses and terms of their original sources. Pretrained models are used under their original access terms, and we do not redistribute their weights. Table 12 summarizes the external datasets and pretrained models used in our experiments, together with the license information stated by the cited sources.

Table 12: External datasets and pretrained models used in the experiments, with license information reported by their sources.

Asset	Type	License / terms of use	Use in this paper
STSA / Stanford Sentiment Treebank	Dataset	License listed as unknown in the SST data card.	Probing / steering benchmark
Cities	Dataset	Underlying GeoNames data: CC BY 4.0. No benchmark-specific license stated in the source repository.	Probing benchmark
Common / CommonClaim	Dataset	Dataset license not specified in the source repository.	Probing benchmark
CounterFact	Dataset	Dataset-specific license not stated on the official download page.	Probing / steering benchmark
HateXplain	Dataset	CC BY 4.0 in the data-card metadata; MIT in the same card's licensing text.	Probing / steering benchmark
Sarcasm News Headlines	Dataset	CC BY 4.0, as listed in the author's Kaggle release.	Probing / steering benchmark
Llama-3.1-8B-Instruct and Llama-3.1-70B-Instruct	Pretrained models	Llama 3.1 Community License Agreement and accompanying Acceptable Use Policy.	Representation extraction
Qwen2.5-3B-Instruct	Pretrained model	Qwen Research License Agreement.	Representation extraction
Qwen2.5-7B-Instruct and Qwen2.5-32B-Instruct	Pretrained models	Apache License 2.0.	Representation extraction
Gemma-7B-it	Pretrained model	Gemma Terms of Use.	Representation extraction

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 and introduction state the algorithmic contribution, the evaluation axes, and the high-dimensional theory, and the claims are supported by the experiments in Section 3 and the analysis in Section 4.

   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: Section 5 discusses the main limitations, including the stylized Gaussian theory and the need to extend the analysis to more realistic feature dependencies; the experimental scope and compute behavior are described in Section 3 and Appendix B.

   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: [Yes]

Justification: The assumptions for the theoretical statements are stated in Section 4, and Appendix A provides the CGMT-based derivation and references the prior ridge-logistic characterization on which the adapted theorem is based.

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: Sections 2–3 and Appendix B specify the probe construction, train/validation/test protocol, hyperparameter selection, baselines, steering protocol, and additional diagnostic experiments needed to reproduce the main results.

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: [Yes]

   Justification: Code and instructions for reproducing the main experimental results, including the five-seed benchmark and added baselines, are publicly available at `https://github.com/Ziqi-Gao/RAPTOR`. The datasets and pretrained models are available from their original sources.

   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: The paper specifies the data splits, layerwise training protocol, train-only standardization, ridge grid and validation tuning, baseline protocols, and steering hyperparameters in Sections 2–3 and Appendix B.

   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: Main probe-accuracy results are summarized over five outer split seeds, with sample standard deviations reported in Appendix B. Appendix B.2 also reports a 95% confidence interval for a seed-paired RAPTOR-minus-xRFM comparison. Directional robustness is reported descriptively.

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: The paper specifies the compute used for the main experiment grid: one NVIDIA H100 GPU was allocated per job and each model–dataset–layer probe fit used one CPU worker. Figure 4 reports wall-clock per-layer probe training time across the full grid, and Appendix B.4 gives the steering-evaluation settings.

   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 work uses existing public models and datasets, does not collect new human-subject data, and is intended for analysis and controlled steering evaluation; we are not aware of any aspect that violates 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: Section 5 includes a broader-impact discussion: the method is intended to support analysis and controllability of language-model representations, while the same capability could be misused to manipulate model behavior or amplify undesirable concepts. The paper therefore recommends task-specific safety evaluation and monitoring for downstream 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: The paper does not release new pretrained language models, image generators, scraped datasets, or other high-risk assets requiring additional release safeguards.

    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: [No]

    Justification: We cite the original sources of the datasets, pretrained models, and baseline methods, and summarize the available license and usage information in Table 12. Explicit dataset licenses could not be verified for some benchmarks; missing or inconsistent license information is noted in the table.

    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: [N/A]

    Justification: The paper does not introduce or release new datasets, pretrained models, or other standalone assets at submission time; it introduces a method and reports experiments on existing assets.

    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 does not involve new crowdsourcing experiments or new research with human subjects. It uses existing public benchmark datasets.

    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 does not involve new research with human subjects, so IRB or equivalent approval is not 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: [Yes]

    Justification: The paper describes the LLM-as-judge protocol used for the external steering evaluation in Appendix B.8. LLMs are not part of the proposed probe-training algorithm; any use of LLMs for writing or editing does not affect the core methodology, scientific rigor, or originality of the work.

    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.