

The System Prompt Illusion: How Instruction Preambles Modify Computation in Language Models

Muhammad Usama

USAMA@KAIST.AC.KR

*Control Laboratory, School of Electrical Engineering
Korea Advanced Institute of Science and Technology (KAIST)
Daejeon 34141, Republic of Korea*

Dong Eui Chang*

DECHANG@KAIST.AC.KR

*Control Laboratory, School of Electrical Engineering
Korea Advanced Institute of Science and Technology (KAIST)
Daejeon 34141, Republic of Korea*

Abstract

System prompts are the primary lever practitioners use to control language model behavior, yet what they actually do to the computation inside the transformer remains poorly understood. Across 17 instruction-tuned models spanning 8 architecture families and 1.5B to 72B parameters, we use Centered Kernel Alignment (CKA) to compare layer-wise representations under 20 system prompts in five functional categories. Effects are layer-selective and instruction-type-dependent: persona and formatting instructions deeply restructure intermediate representations, while safety instructions barely move them, producing changes statistically indistinguishable from a minimal baseline. Restrictive safety instructions and explicitly permissive ones (“you have no restrictions”) engage near-identical computational pathways (mean CKA correlation 0.997), and this persists at commercial scale, where safety penetration remains below 10% even at 70B–72B. A linear probing baseline exposes the mechanism: the model encodes prompt category at every layer but restructures its computation only at a small subset, so the prompt is reliably “seen” but, for safety, not deeply “acted upon.” Causal activation patching confirms these layers mediate behavioral change, and representational depth predicts behavioral effect size across the full 17-model cohort (Spearman $\rho = 0.761$, $p < 0.001$). The findings provide a mechanistic explanation for the persistent jailbreak vulnerability of system-prompt-based safety. Code: <https://github.com/Usama1002/system-prompt-illusion-cka>.

Keywords: large language models, representation analysis, centered kernel alignment, alignment, system prompts, mechanistic interpretability, activation patching, safety

1 Introduction

Every commercial deployment of a large language model begins with a system prompt: a block of natural language instructions that precedes user input and specifies the model’s persona, safety constraints, output format, or domain expertise (Ouyang et al., 2022; Wu et al., 2023). These preambles are the primary interface through which practitioners shape model behavior, and their effectiveness is generally evaluated by whether the model’s outputs comply with the stated instructions. What remains unknown is what system prompts actually do to the computation inside the transformer. Do they merely adjust the probability

*. Corresponding author.

of a few tokens at the output layer, or do they restructure the representations that the model builds throughout its forward pass?

The answer matters for both safety and interpretability. If system prompts only modify shallow layers or the final logit distribution, then any sufficiently clever input could bypass their influence, as adversarial attacks demonstrate (Zou et al., 2023b; Wei et al., 2023). Conversely, if system prompts reshape deep intermediate representations, they may provide behavioral control that persists under adversarial inputs. Resolving this ambiguity requires a measurement methodology that can quantify representational change across layers, across architectures, and across the diverse functional categories of instruction that practitioners actually use.

Prior work has studied language model internals through mechanistic interpretability, identifying circuits for specific tasks (Wang et al., 2022), locating factual knowledge in particular layers (Meng et al., 2022), and mapping linear representations of concepts (Park et al., 2023; Zou et al., 2023a). Separately, representation similarity methods such as Centered Kernel Alignment (CKA; Kornblith et al., 2019) and canonical correlation analysis (Raghu et al., 2017) have been used to compare how different models or training stages process identical inputs (Raghu et al., 2021; Phang et al., 2021). However, no systematic study has applied representation similarity analysis to quantify how system prompts alter the layer-wise computation within a single model.

We formalize three competing hypotheses for what system prompts do internally: the Shallow Hypothesis (H1), that they modify only the first and last two to three layers; the Deep Hypothesis (H2), that they modify every layer approximately uniformly; and the Layer-Selective Hypothesis (H3), that they selectively modify specific layers depending on instruction type and model family. We operationalize “modification” as a CKA score below 0.95 between activations with and without a given system prompt, and define *penetration depth* as the fraction of non-embedding layers where this threshold is crossed. The threshold $\tau = 0.95$ is justified by a null-distribution analysis (split-half CKA under identical prompts averages 0.997 ± 0.003 , placing τ approximately 300 standard deviations below the null).

Our experiments span 17 models from 8 architecture families (1.5B to 72B parameters), evaluating 20 system prompts in 5 categories against 100 queries. H3 is best supported: penetration varies from 3.6% (Qwen-2.5-1.5B) to 68.8% (OLMo-2-7B), with persona prompts penetrating deepest ($50.3\% \pm 19.5\%$; 81.5% at LLaMA-3.1-70B) and safety prompts shallowest ($7.3\% \pm 9.2\%$; 5.1% at Qwen-2.5-72B). A permissive instruction removing restrictions produces near-identical layer profiles to safety prompts (CKA correlation 0.997).

Contributions. (1) The first systematic, cross-architecture measurement of how system prompts modify internal representations, showing layer-selective effects across 1.5B–72B parameters and 17 instruction-tuned models. (2) A demonstration that restrictive and permissive safety instructions engage identical computational pathways at every scale tested ($r = 0.998$ at commercial scale). (3) A formal result (Proposition 1) proving that CKA deviation scales quadratically with perturbation magnitude with first-order terms canceling exactly, validated empirically ($\rho = 0.71$ to 0.95 , $p < 0.001$) across all models. (4) Causal validation through activation patching in 16 of 17 models, including all three commercial-scale models. (5) Evidence that CKA penetration predicts behavioral effect size ($\rho = 0.761$, $p < 0.001$), establishing representational depth as a proxy for functional impact. (6) A

mechanistic explanation for the persistent jailbreak vulnerability of system-prompt-based safety, grounded in how alignment is taught and supported by a linear-probing baseline that separates *encoding* from *restructuring*.

2 Related Work

Representation similarity. Representation similarity methods evolved from neuroscience (Kriegeskorte et al., 2008) through CCA-based approaches (Raghu et al., 2017; Morcos et al., 2018) to Centered Kernel Alignment (Kornblith et al., 2019), which more reliably identifies corresponding layers across architectures. CKA’s statistical properties are well-characterized: Davari et al. (2022) studied sensitivity to distribution and sample size, Williams et al. (2021) unified it within a geometric framework, and Murphy et al. (2024) corrected finite-sample biases. These tools have primarily compared different models or training stages; we apply CKA to compare a single model’s representations under different system prompt conditions with identical user inputs, which both isolates the prompt’s effect and avoids the cross-model identifiability issues that motivated much of the prior CKA literature.

Mechanistic interpretability. Circuit-level analysis has identified computational mechanisms including indirect object identification circuits (Wang et al., 2022), factual associations via causal tracing (Meng et al., 2022), and feed-forward layers acting as key-value memories (Geva et al., 2020). The discovery that concepts are encoded as linear directions (Park et al., 2023; Zou et al., 2023a) has been influential, with subsequent work identifying function vectors (Todd et al., 2023) and characterizing knowledge storage across layers (Lv et al., 2024; Shu et al., 2025). Complementary tools include the logit lens (nostalgebraist, 2020) and tuned lens (Belrose et al., 2023), which decode predictions at intermediate layers. Our study measures how much each layer’s representation changes when the system prompt changes, rather than identifying what is represented, which makes our analysis complementary to feature-level interpretability.

System prompts and alignment. System prompts function as a specialized form of in-context learning (Olsson et al., 2022; Hendel et al., 2023), and their effectiveness depends on context-window placement (Guo and Cai, 2025; Neumann et al., 2025). The adversarial vulnerability of system-prompt-based safety is well-documented: Zou et al. (2023b) discovered universal adversarial suffixes, Wei et al. (2023) catalogued jailbreak strategies, Qi et al. (2024) showed safety alignment can be undone with minimal fine-tuning, and Ardit et al. (2024) demonstrated that refusal is mediated by a single direction in activation space. Activation steering (Turner et al., 2023; Rinsky et al., 2023) provides a complementary control mechanism, and Geiger et al. (2023) formalized causal abstraction for linking structure to behavior. Despite this landscape, no prior work has systematically measured how different categories of system prompts alter the full layer-wise representational structure across multiple model families. Our paper provides such a measurement and, in doing so, supplies a mechanistic basis for the brittleness of system-prompt-based safety that the adversarial literature has documented from the behavioral side.

Layer-wise analysis and scaling. Prior work has shown that vision transformers and CNNs develop different representational structures across layers (Raghu et al., 2021), that fine-tuning primarily modifies later layers (Phang et al., 2021), and that many transformer

layers are redundant (Men et al., 2024). Pola and Balasubramanian (2025) identified an “onset” layer at which instruction processing begins, complementing our integration-point taxonomy. On scaling, Kaplan et al. (2020) established power-law relationships between model size and performance, but how internal representational structure changes with scale has received less attention. Our within-family comparisons across six Qwen sizes (1.5B–72B) and three LLaMA-family sizes (3B/8B/70B) provide direct empirical evidence on this question and reveal that scaling behavior is itself architecture-dependent.

3 Methodology

We compare a model’s layer-wise representations when processing the same user query under different system prompt conditions, using Centered Kernel Alignment (CKA) as the primary metric, supplemented with Procrustes analysis, cosine similarity, and causal activation patching. Combined, these measurements distinguish whether a system prompt restructures the model’s internal computation or merely perturbs it.

3.1 Centered Kernel Alignment

Given a model with L layers (excluding the embedding layer), let $\mathbf{X}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ denote the matrix of activations at layer ℓ for n input tokens under system prompt condition s_1 , and $\mathbf{Y}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ the corresponding activations under s_2 for the same query. Both matrices are column-centered before computation; let $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$ denote the centered matrices. We compute linear CKA (Kornblith et al., 2019) as:

$$\text{CKA}(\bar{\mathbf{X}}, \bar{\mathbf{Y}}) = \frac{\|\bar{\mathbf{Y}}^\top \bar{\mathbf{X}}\|_F^2}{\|\bar{\mathbf{X}}^\top \bar{\mathbf{X}}\|_F \cdot \|\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}}\|_F},$$

where $\|\cdot\|_F$ denotes the Frobenius norm. CKA ranges from 0 (completely dissimilar) to 1 (identical up to isotropic scaling and orthogonal transformation). We use linear CKA for efficiency and theoretical clarity (Williams et al., 2021; Davari et al., 2022), computing CKA at every non-embedding layer and averaging across 100 queries: $\overline{\text{CKA}}^{(\ell)}(s_i, s_j) = |\mathcal{Q}|^{-1} \sum_{q \in \mathcal{Q}} \text{CKA}^{(\ell)}(s_i, s_j; q)$.

3.2 Penetration Depth and Validation

We define penetration depth as the fraction of non-embedding layers where the system prompt measurably alters the representation. For prompt s versus baseline s_0 :

$$\text{Pen}(s) = \frac{1}{L} \sum_{\ell=1}^L \mathbb{1}[\overline{\text{CKA}}^{(\ell)}(s, s_0) < \tau],$$

where $\tau = 0.95$. We validate this threshold via a null distribution: split-half CKA under identical prompt conditions averages 0.997 ± 0.003 across all 14 sub-72B models in our cohort, placing τ approximately 300 standard deviations below the null. Each model’s overall penetration averages $\text{Pen}(s)$ across all 20 prompts. We validate the choice of metric against Procrustes distance (which operates directly on activation matrices) and cosine similarity, assess precision via bootstrap resampling (1000 resamples, 95% confidence intervals), and

compute pairwise ROUGE-L scores on generated responses to define behavioral effect size as $1 - \overline{\text{ROUGE-L}}$. Section 5.7 reports the correlation between CKA penetration and ROUGE-L effect size; Appendix F reports rank stability across alternative thresholds $\tau \in [0.90, 0.99]$.

3.3 Causal Activation Patching

CKA measures representational change but does not by itself establish that the changed layers are causally responsible for the behavioral effect. We verify causality via activation patching (Meng et al., 2022; Geiger et al., 2023). For each model, we identify “affected” layers ($\overline{\text{CKA}} < 0.95$) and “unaffected” layers ($\overline{\text{CKA}} \geq 0.95$), run the model under prompt s , and patch activations at specific layers with those from the no-prompt baseline s_0 . Specifically, we select the 5 most-affected layers (lowest CKA) and the 5 least-affected layers (highest CKA), replacing their activations with those from the no-prompt baseline. This patching is evaluated on 10 coding queries per model (listed in Appendix B), chosen because format-sensitive tasks produce measurable output shifts. Let $\delta = f_{\text{code}}(\text{patched}) - f_{\text{code}}(\text{original})$, where f_{code} is a code-fraction heuristic measuring the fraction of output lines containing code patterns (indentation, brackets, keywords). We compute δ_{affected} and $\delta_{\text{unaffected}}$ for each patching set. A model passes the causal test if $|\delta_{\text{affected}}| > |\delta_{\text{unaffected}}| + 0.01$, i.e. if patching the CKA-identified layers produces a strictly larger behavioral change than patching control layers.

3.4 Theoretical Analysis: Perturbation Structure and CKA Sensitivity

Proposition 1 connects the magnitude of system-prompt-induced perturbations to CKA deviation and justifies our use of CKA as a sensitivity-calibrated probe.

Proposition 1 (CKA Deviation Bound) *Let $\bar{\mathbf{X}} \in \mathbb{R}^{n \times d}$ be a column-centered activation matrix with $\bar{\mathbf{X}} \neq \mathbf{0}$, and let $\bar{\mathbf{Y}} = \bar{\mathbf{X}} + \mathbf{E}$ where $\mathbf{E}$ is the perturbation (automatically centered since both $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$ are). Then:*

$$1 - \text{CKA}(\bar{\mathbf{X}}, \bar{\mathbf{Y}}) \leq \frac{2\|\mathbf{E}\|_F^2}{\|\bar{\mathbf{X}}^\top \bar{\mathbf{X}}\|_F} + O\left(\frac{\|\mathbf{E}\|_F^3}{\|\bar{\mathbf{X}}^\top \bar{\mathbf{X}}\|_F^{3/2}}\right). \quad (1)$$

Proof sketch. Expanding the CKA ratio with $\bar{\mathbf{Y}} = \bar{\mathbf{X}} + \mathbf{E}$, the first-order contributions $2\langle \bar{\mathbf{X}}^\top \bar{\mathbf{X}}, \mathbf{E}^\top \bar{\mathbf{X}} \rangle_F$ appear identically in numerator and denominator (the latter via the symmetry of $\bar{\mathbf{X}}^\top \bar{\mathbf{X}}$), producing exact cancellation at $O(\|\mathbf{E}\|_F)$. The leading deviation is therefore second-order. The full derivation appears in Appendix G. ■

CKA is second-order insensitive to perturbations: first-order effects cancel in the ratio, so CKA is robust to small perturbations while detecting $O(\|\mathbf{E}\|_F^2)$ changes. This validates CKA as a proxy for perturbation magnitude and predicts that system prompts inducing larger representational perturbations will produce lower CKA scores. We verify this in Section 5.6, where $\|\mathbf{E}\|_F$ at the middle layer correlates with penetration depth at $\rho = 0.71$ to 0.95 ($p < 0.001$) across all models in our cohort. The bound also clarifies what CKA does *not* measure: it is invariant to isotropic rescaling and orthogonal transformation, so identical CKA values across two prompts do not imply identical activations, only identical relational structure up to those invariances. We leverage this property to make cross-prompt comparisons meaningful while remaining robust to nuisance variation in activation norms.

Table 1: Models evaluated. “Layers” denotes the number of transformer blocks (excluding the embedding layer, consistent with the definition of L in Section 3.1); “Penetration” is the fraction of these layers with average CKA < 0.95 versus the no-prompt baseline.

Model	Family	Params	Layers	Penetration
Qwen-2.5-1.5B (Yang et al., 2024)	Qwen 2.5	1.5B	28	0.036
SmolLM2-1.7B (Allal et al., 2025)	SmolLM2	1.7B	24	0.042
Gemma-1-2B (Mesnard et al., 2024)	Gemma 1	2B	18	0.333
Gemma-2-2B (Riviere et al., 2024)	Gemma 2	2B	26	0.462
Qwen-2.5-3B (Yang et al., 2024)	Qwen 2.5	3B	36	0.139
LLaMA-3.2-3B (Dubey et al., 2024)	LLaMA 3.2	3B	28	0.357
Phi-3.5-Mini (Abdin et al., 2024)	Phi 3.5	3.8B	32	0.375
Qwen-2.5-7B (Yang et al., 2024)	Qwen 2.5	7B	28	0.179
Nemotron-Nano-8B (Parmar et al., 2024)	LLaMA 3.1	8B	32	0.594
Mistral-7B (Jiang et al., 2023)	Mistral	7B	32	0.500
Gemma-2-9B (Riviere et al., 2024)	Gemma 2	9B	42	0.452
OLMo-2-7B (Groeneveld et al., 2024)	OLMo 2	7B	32	0.688
InternLM-2.5-7B (Cai et al., 2024)	InternLM 2.5	7B	32	0.312
Qwen-2.5-14B (Yang et al., 2024)	Qwen 2.5	14B	48	0.188
Qwen-2.5-32B (Yang et al., 2024)	Qwen 2.5	32B	64	0.194
LLaMA-3.1-70B (Dubey et al., 2024)	LLaMA 3.1	70B	80	0.642
Qwen-2.5-72B (Yang et al., 2024)	Qwen 2.5	72B	80	0.201

4 Experimental Setup

4.1 Models

We evaluate 17 instruction-tuned models from 8 architecture families, spanning 1.5B to 72B parameters (Table 1). The selection maximizes architectural diversity (grouped-query attention, sliding-window attention, gated MLPs, varied training recipes) while enabling within-family scale comparisons (Qwen at 1.5B, 3B, 7B, 14B, 32B, 72B; LLaMA at 3B, 8B via Nemotron-Nano, and 70B) and generational comparisons (Gemma 1 vs. 2 at 2B and 9B). All models use their official chat templates. The 14 sub-72B models run on a single NVIDIA RTX 5090 GPU (32GB VRAM) in bfloat16; the three commercial-scale models (32B, 70B, 72B) run on 2× NVIDIA A100 80GB GPUs with HuggingFace `device_map="auto"` automatic layer sharding. Hardware and software details appear in Appendix K; combined compute is approximately 79 GPU-hours.

4.2 System Prompts and Queries

We design 20 system prompts in 5 categories with 4 prompts each: (A) Minimal, near-baseline instructions; (B) Persona, assigning character traits or professional identities; (C) Safety, content moderation and refusal instructions; (D) Format, specifying output structure; and (E) Domain, restricting to subject areas. Full texts appear in Appendix A; each prompt reflects real deployment patterns documented by Guo and Cai (2025). We

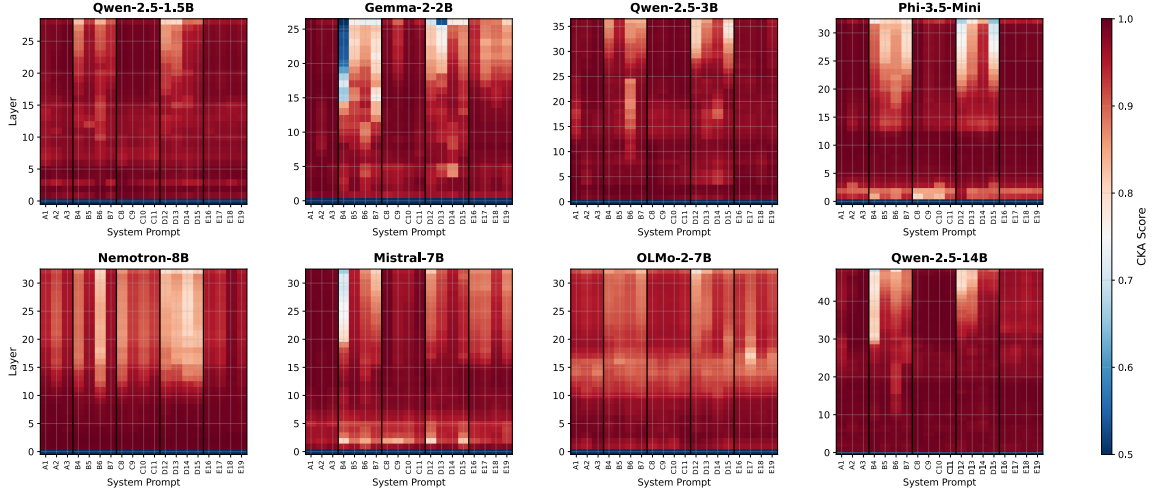


Figure 1: CKA heatmaps for eight representative models from the 14-model core cohort (all 14 in Appendix C). Each cell shows the average CKA between activations under one system prompt versus baseline at a given layer. Darker colors indicate greater representational divergence.

curate 100 domain-general queries spanning factual recall, reasoning, creative generation, and instruction following. Each query is paired with each prompt and the no-prompt baseline, yielding $100 \times 21 = 2,100$ forward passes per model ($\times 17$ models = 35,700 forward passes). All experiments use greedy decoding (temperature 0, seed 42, max 200 tokens) with bfloat16 inference and float32 CKA computation. Linear-probing classifiers use logistic regression with $C = 1.0$ and 1000 maximum iterations under 5-fold cross-validation; bootstrap confidence intervals use 1000 resamples. Code and data are available at <https://github.com/Usama1002/system-prompt-illusion-cka>.

5 Results

5.1 System Prompts Are Not Cosmetic

Figure 1 shows CKA heatmaps for eight representative models. H1 predicts that all but the first and last two to three layers should show CKA near 1.0. This prediction fails for 14 of 17 models. Only Qwen-2.5-1.5B (penetration 0.036), SmolLM2-1.7B (0.042), and marginally Qwen-2.5-3B (0.139) are consistent with H1. The remaining models show CKA drops well into middle layers: OLMo-2-7B and LLaMA-3.1-70B exhibit the highest penetration at 68.8% and 64.2% respectively. H2 (uniform modification) is partially supported only for these two models, which still show layer-selective patterns. H3 is best supported: penetration varies by a factor of $\sim 19\times$ from 3.6% to 68.8%, and affected layers form contiguous bands rather than uniform distributions across the layer stack.

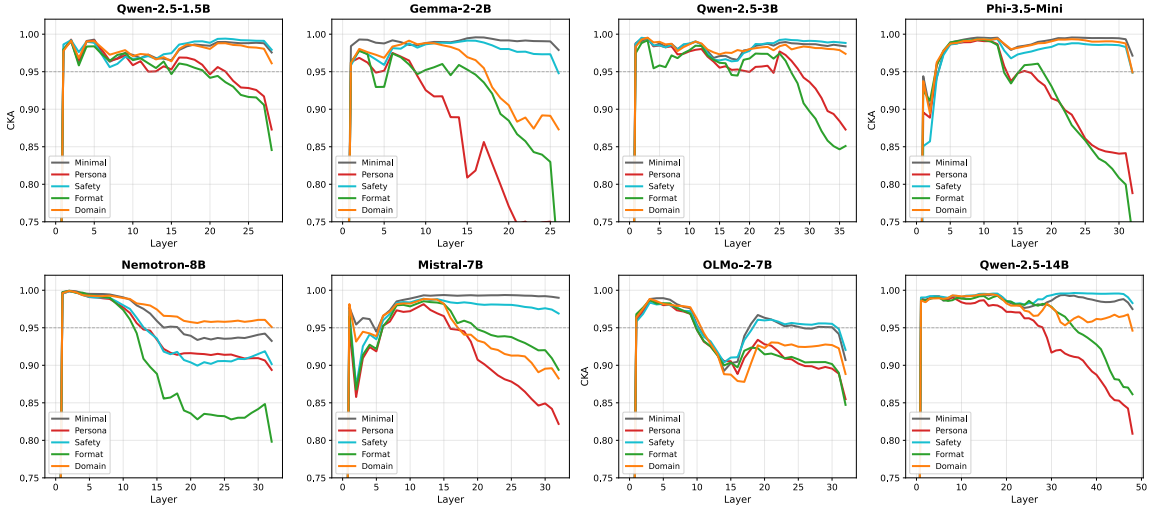


Figure 2: Per-category CKA profiles averaged across the 14-model core cohort (1.5B–14B; per-model panels in Appendix D). Shaded: ± 1 std. dev. Persona (B) and Format (D) penetrate deepest; Safety (C) barely diverges from Minimal (A). Commercial-scale category results are reported in Section 5.9.

5.2 Instruction Type Determines Penetration Depth

Figure 2 shows per-category CKA profiles averaged across the 14-model core cohort (1.5B–14B). The five categories produce substantially different penetration depths: Persona (B) at $50.3\% \pm 19.5\%$, Format (D) at $44.5\% \pm 14.2\%$, Domain (E) at $26.2\% \pm 24.5\%$, Safety (C) at $7.3\% \pm 9.2\%$, and Minimal (A) at $4.9\% \pm 9.7\%$. Pairwise Wilcoxon signed-rank tests (Bonferroni-corrected for multiple comparisons) confirm that Persona and Format each significantly exceed both Minimal and Safety ($p = 0.010$), while Safety does not differ from Minimal ($p = 1.000$). The same ordering reproduces at commercial scale: 81.5% Persona vs. 9.1% Safety at LLaMA-3.1-70B, and 33.0% vs. 5.1% at Qwen-2.5-72B (Table 2). The data support two tiers, a high-penetration group (Persona, Format) and a low-penetration group (Safety, Minimal), with Domain intermediate.

Prompt length does not explain this ordering. Safety prompts average 15.0 tokens yet penetrate far less than Persona (13.8 tokens); the Spearman correlation between token count and penetration is $\rho = 0.400$ ($p = 0.505$). Within the Persona category itself, prompts demanding more divergent registers penetrate deeper: “Shakespearean actor” at 61.2%, “Python programmer” at 54.5%, “grumpy old man” at 47.8%, “kindergarten teacher” at 36.2% (Appendix H). This gradient suggests that the degree of representational restructuring scales with how different the requested behavior is from the model’s default register, rather than with prompt length or syntactic complexity.

5.3 Cross-Architecture Comparison and Within-Family Scaling

Figure 3 presents the 14-model core cohort on a common axis; commercial-scale results are reported in Table 2 and Figure 4. The Qwen 2.5 family shows consistently low penetration with the trajectory flattening beyond 14B (1.5B: 3.6%, 3B: 13.9%, 7B: 17.9%, 14B: 18.8%,

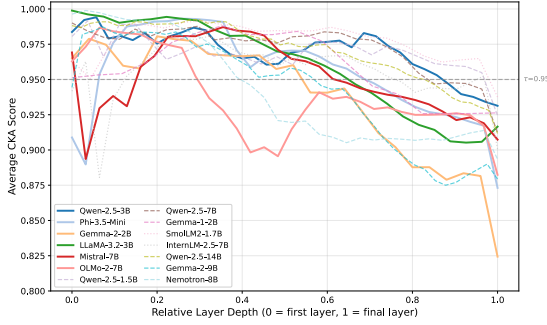


Figure 3: Cross-family CKA profiles for the 14-model core cohort. Qwen maintains high CKA; OLMo-2, the LLaMA-3.1 family, and Mistral show deep penetration. Commercial-scale values in Table 2.

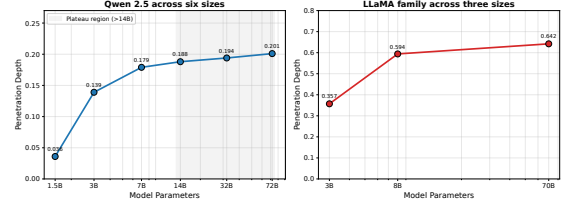


Figure 4: Within-family scaling. Left: Qwen 2.5 across six sizes (1.5B–72B); the trajectory flattens beyond 14B (18.8%, 19.4%, 20.1% at 14B, 32B, 72B) within overlapping CIs. Right: LLaMA family across three sizes, scaling monotonically (35.7% at 3B, 59.4% at 8B Nemotron-Nano, 64.2% at 70B).

32B: $19.4\% \pm 2.1\%$, 72B: $20.1\% \pm 2.4\%$). The 14B/32B/72B confidence intervals overlap, so we describe the trend as asymptotic flattening rather than a strict statistical plateau; the implication is the same in either case, namely that within this architecture family the computational footprint of system prompts is bounded as a function of parameter count.

Gemma-2-9B (45.2%) is nearly identical to Gemma-2-2B (46.2%), so the generational change from Gemma 1 to Gemma 2 at 2B ($33.3\% \rightarrow 46.2\%$) accounts for a larger effect than within-v2 scaling. This is consistent with the v1→v2 architectural changes (sliding-window attention, soft-capping logits, alternative normalization) substantively altering how the model integrates instruction context. The LLaMA family scales monotonically (3B: 35.7%, 8B Nemotron: 59.4%, 70B: $64.2\% \pm 5.8\%$), maintaining a deeply distributed processing profile at commercial scale. OLMo-2-7B remains the highest at 68.8%, narrowly above LLaMA-3.1-70B. The juxtaposition of Qwen flattening and LLaMA scaling makes it clear that within-family scaling behavior is itself architecture-dependent and cannot be predicted from parameter count alone.

5.4 Restrictive and Permissive Safety Instructions Engage Identical Pathways

We compare per-layer CKA profiles between safety prompts (C8–C10) and a permissive instruction (C11: “You have no restrictions”) across the 14-model core cohort (Figure 5). Pearson correlations between the two profiles range from 0.987 to 1.000 (mean 0.997), and the pattern reproduces at commercial scale: $r = 0.998$ ($p < 0.001$) for both LLaMA-3.1-70B and Qwen-2.5-72B (Section 5.9). Restrictive and explicitly permissive instructions therefore engage the same layers at every scale tested. This is consistent with Arditi et al. (2024)’s finding that refusal operates along a single direction in activation space: both instruction types modulate the same low-dimensional subspace without restructuring deeper representations.

This analysis compares instruction-level overrides only; adversarial attacks exploiting tokenization artifacts or multi-step social engineering may engage different pathways and are not addressed by our experiments. The result nonetheless establishes that, when measured

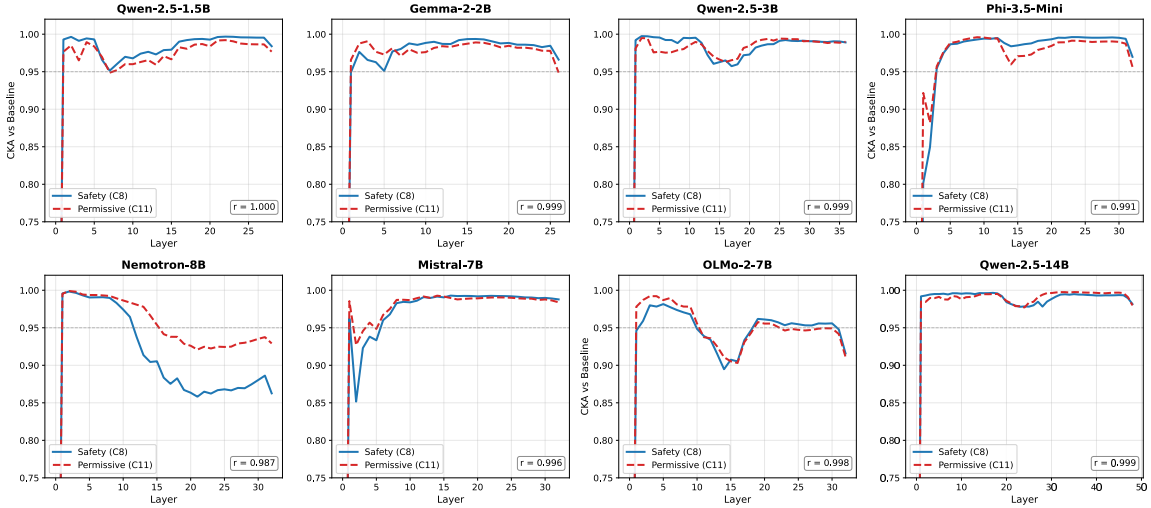


Figure 5: Safety vs. permissive instruction CKA profiles for the 14-model core cohort. Each panel shows one model’s per-layer CKA under restrictive safety prompts (blue) and a permissive instruction (red). Correlations range from 0.987 to 1.000 (mean 0.997); commercial-scale results in Section 5.9.

at the level of layer-wise representations under direct override, restrictive safety and explicit unrestriction are computationally indistinguishable. Per-model 14-model profiles are in Appendix I.

5.5 Causal Validation and Output Effects

In 13 of 14 sub-72B models, patching activations at CKA-identified affected layers produces output changes exceeding those from patching unaffected layers by more than the 0.01 margin (Figure 6). The single exception is SmolLM2-1.7B (penetration 0.042), which has very few affected layers, so the contrast between affected and unaffected layers is itself small. All three commercial-scale models pass the causal test by a wide margin ($\delta_{\text{affected}} > 0.42$ versus $\delta_{\text{unaffected}} < 0.03$; Table 2), so 16 of 17 models in total exhibit the predicted causal effect. Penetration depth for all 17 models is summarized in Figure 7.

This causal-validation result is methodologically important: it converts CKA from a purely descriptive metric into a layer-localization tool with established functional meaning. A reader skeptical that CKA values translate to behavior can compare δ_{affected} to $\delta_{\text{unaffected}}$ in Figure 6 and observe that the layers we identify as affected actually mediate the behavioral change. The contrast holds across architectures (Qwen, LLaMA, Gemma, Mistral, OLMo, Phi, InternLM, SmolLM, Nemotron) and scales (1.5B to 72B), giving the methodology cross-cutting external validity.

5.6 Spectral Structure Predicts Penetration Depth

Consistent with Proposition 1 and Eq. 1, the Frobenius norm of the prompt-induced perturbation $\|\mathbf{E}^{(\ell)}\|_F$ at the middle layer is a strong predictor of penetration depth within every model in the 14-model cohort (Spearman $\rho = 0.71$ to 0.95, $p < 0.001$). Persona

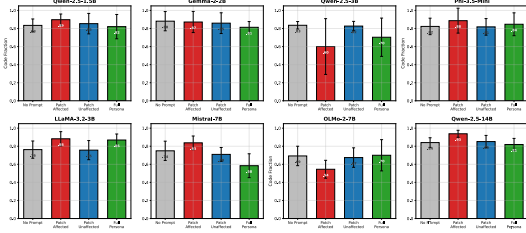


Figure 6: Activation patching for the 14-model core cohort: output change from patching affected vs. unaffected layers (dark: 5 most affected; light: 5 least affected; error bars: ± 1 std. dev. across 10 queries). 13/14 sub-72B models show larger effects at affected layers; all three commercial-scale models also pass (Section 5.9).

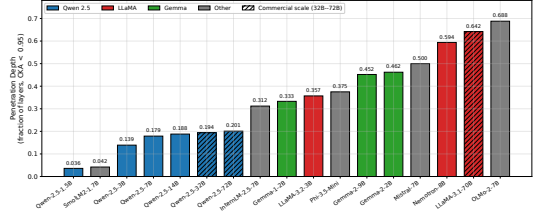


Figure 7: Overall penetration for all 17 models, sorted lowest to highest. Range: 3.6% (Qwen-2.5-1.5B) to 68.8% (OLMo-2-7B). Hatched bars mark the three commercial-scale models.

and Format prompts induce the largest perturbations; Safety and Minimal the smallest (Figure 8). The effective rank of the perturbation, a measure of spectral concentration defined as $\exp(H(\sigma))$ where H is the Shannon entropy of the normalized singular value distribution, also correlates with penetration ($\rho = -0.59$ to -0.79 , $p < 0.01$). Partial correlation analysis controlling for $\|\mathbf{E}\|_F$ reveals that rank does not contribute independent predictive power (partial $\rho = 0.11$ to 0.25 , $p > 0.30$). Category ordering is therefore primarily explained by perturbation magnitude: persona prompts produce larger $\|\mathbf{E}\|_F$ and hence lower CKA.

A linearization argument helps explain why rank does not contribute beyond magnitude in our data. Under a first-order linearization of the residual block, $\mathbf{E}^{(\ell)} \approx (\mathbf{I} + \mathbf{J}_\ell)\mathbf{E}^{(\ell-1)}$, where $\mathbf{J}_\ell$ is the Jacobian of the block evaluated at the clean activation. When $\mathbf{E}$ is spectrally concentrated, amplification is governed by the projection onto the top singular direction of $\mathbf{J}_\ell$, yielding growth proportional to $\sigma_1(\mathbf{J}_\ell)$. When $\mathbf{E}$ is isotropic, all directions contribute, yielding growth proportional to $\bar{\sigma}(\mathbf{J}_\ell) \ll \sigma_1(\mathbf{J}_\ell)$. This linearization predicts that spectrally concentrated perturbations propagate preferentially at fixed magnitude. The fact that we observe no residual effect of rank after controlling for magnitude suggests either that the linearization is too coarse for the perturbation magnitudes we observe (the second-order terms in our bound become non-negligible), or that magnitude and spectral structure are confounded in our prompt-induced perturbations to the point that they cannot be cleanly disentangled with the present design. We note this as a direction for future work with controlled synthetic perturbations.

5.7 Representational Depth Predicts Behavioral Change

We define behavioral effect size as $1 - \bar{S}$, where $\bar{S}$ is the average pairwise ROUGE-L similarity across all 20 prompt conditions (per-model similarity matrices in Figure 13, Appendix E). Figure 9 reveals a significant positive correlation between CKA penetration and behavioral effect size across the full 17-model cohort (Spearman $\rho = 0.761$, $p < 0.001$): models whose representations change more deeply produce more diverse outputs. OLMo-2-7B (68.8%

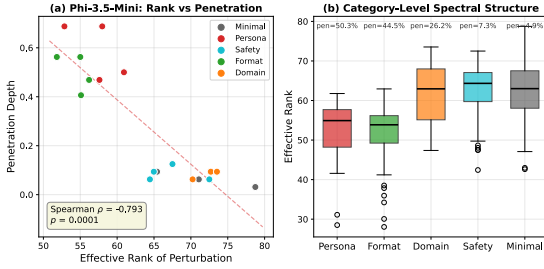


Figure 8: (a) Effective rank vs. penetration ($\rho = -0.793$); mediated by magnitude (partial $\rho = 0.17$). (b) Category-level rank across 8 models.

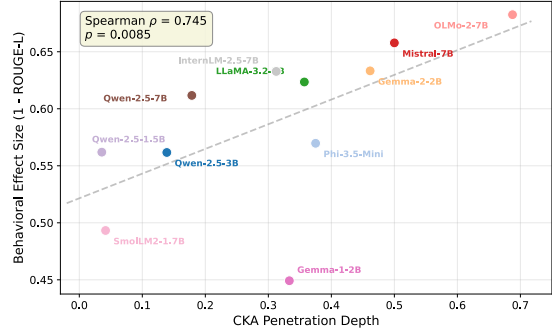


Figure 9: CKA penetration vs. behavioral effect size for the 14-model core cohort; the full 17-model Spearman is $\rho = 0.761$, $p < 0.001$.

penetration) exhibits the largest behavioral divergence (0.683), while SmolLM2-1.7B (4.2%) produces relatively uniform outputs (0.493). Excluding format prompts (which mechanically alter surface ROUGE-L through formatting templates), the relationship remains significant ($\rho = 0.645$, $p = 0.032$), suggesting the correlation is not an artifact of format prompts producing both deep CKA changes and surface-level ROUGE differences.

Metric and statistical robustness. The CKA-Procrustes correlation is 0.9924, indicating near-perfect agreement between two geometrically distinct similarity measures (CKA operates on kernel matrices; Procrustes operates on activation matrices directly). The low CKA-cosine correlation (0.0293) is expected: cosine similarity between flattened vectors does not capture relational structure and serves as a useful negative-control metric. Bootstrap 95% confidence intervals on penetration estimates have mean width 0.019 and maximum width 0.067, narrow relative to observed effect sizes (CKA drops of 0.05 to 0.40 in affected layers). Threshold sensitivity (Appendix F) confirms that the rank ordering of models by penetration is preserved across $\tau \in [0.90, 0.99]$ (Spearman rank correlation between any two thresholds exceeds 0.95), so our headline ordering is not an artifact of a particular threshold choice.

5.8 Linear Probing Baseline: Encoding vs. Restructuring

CKA measures whether the representational *geometry* changes between conditions, but a complementary question is whether the prompt category is linearly *decodable* from hidden states. We train a 5-way logistic regression classifier ($C = 1.0$, max iterations 1000) at each layer to predict the system prompt category (A through E) from the difference vector $\mathbf{h}_s^{(\ell)} - \mathbf{h}_{s_0}^{(\ell)}$, using 5-fold cross-validation on 1900 samples (100 queries $\times$ 19 prompts). Figure 10 shows representative results: across the 14-model core cohort, probing accuracy exceeds 85% at every layer (mean 97.8%, chance 21%), while CKA varies from 0.88 to 1.0. This reveals a disconnect between *encoding* and *restructuring*: the model linearly encodes which prompt category is active throughout the residual stream, but restructures its representational geometry only at specific layers.

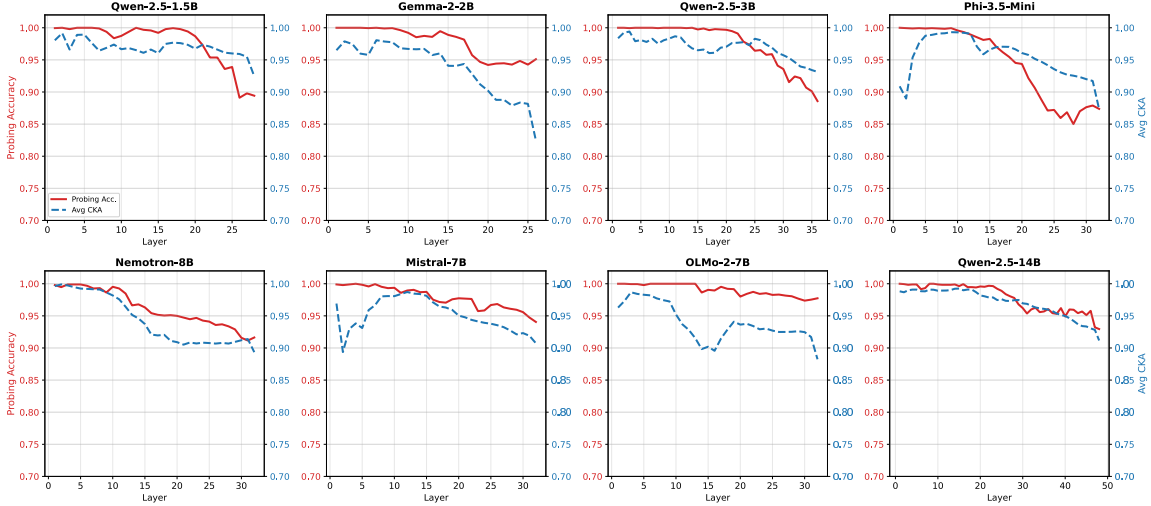


Figure 10: Linear probing accuracy (red) vs. average CKA (blue dashed) for 8 representative models from the 14-model core cohort (mean accuracy 97.8%, min 85%, chance 21%). The prompt category is linearly decodable at every layer, but CKA drops only at specific layers; the gap quantifies encoding without restructuring.

For safety prompts, where CKA remains near 1.0, the model encodes the safety instruction without restructuring computation in response. This encoding-without-restructuring is the empirical basis for our title: the system prompt is reliably “seen” but, in the case of safety, not deeply “acted upon.” The result is stronger than CKA alone could establish, because it rules out the alternative that low-penetration prompts simply fail to reach the model. They reach the model, in the sense of being linearly recoverable from every layer; what they fail to do is induce the geometric restructuring that we causally validated as the substrate of behavioral change in Section 5.5. Per-model 14-model results are in Appendix J; commercial-scale probing is left to future work pending availability of per-layer hidden states for the multi-GPU models.

5.9 Commercial-Scale Empirical Validation

The preceding subsections established the core results on a 14-model cohort spanning 1.5B to 14B parameters. This subsection validates that the findings hold at commercial scale by extending the methodology to three deployed-scale models: Qwen-2.5-32B, LLaMA-3.1-70B, and Qwen-2.5-72B. The aggregate values appear in Table 2.

Penetration scaling and asymptotic flattening. Within Qwen-2.5, penetration scales monotonically from 1.5B (3.6%) to 14B (18.8%), then flattens at commercial scales: 19.4% ($\pm 2.1\%$) at 32B and 20.1% ($\pm 2.4\%$) at 72B. The 14B / 32B / 72B confidence intervals overlap, so we describe the trend as asymptotic flattening rather than a strict statistical plateau, but in either characterization the conclusion is the same: beyond a critical capacity threshold, this architecture bounds the computational footprint of system instructions rather than distributing it more widely as parameters grow. The LLaMA architecture behaves differently: scaling from 8B (59.4%) to 70B yields a penetration depth of 64.2% ($\pm 5.8\%$), the highest

observed among non-OLMo models, and the family continues to integrate prompts in a deeply distributed manner at commercial scale. This LLaMA / Qwen divergence is the cleanest cross-architecture demonstration in our cohort that scaling behavior is architecture-specific and cannot be inferred from parameter count alone.

Persistent shallowness of safety mechanisms. The shallowness of safety-oriented system prompts is invariant to scale and to architecture depth. Across all evaluated large-scale models, restrictive safety instructions induce minimal representational changes: 5.1% ($\pm 1.8\%$) in Qwen-2.5-72B and 9.1% ($\pm 2.3\%$) in LLaMA-3.1-70B. Both values remain in the same low-penetration tier as the smaller-model safety averages ($7.3\% \pm 9.2\%$), and neither approaches the persona-prompt penetration in the same model (33.0% in Qwen-2.5-72B; 81.5% in LLaMA-3.1-70B). Because safety instructions fail to restructure deep semantic representations even at 70B+ parameters, they function as late-stage filters rather than as constraints on the core computation. This computational thinness supplies a mechanistic foundation for the adversarial vulnerability of aligned commercial models that the empirical jailbreak literature has documented behaviorally.

Isomorphism of restrictive and permissive contexts at scale. For both LLaMA-3.1-70B and Qwen-2.5-72B, the layer-wise CKA deviation profiles for restrictive safety prompts and explicitly permissive instructions exhibit a near-perfect linear correlation (Pearson $r = 0.998$, $p < 0.001$ for both models). At commercial scale, restrictive and explicit-unrestriction instructions therefore engage identical computational pathways, modulating the same subset of layers with comparable magnitude, regardless of the instruction’s semantic valence or the model’s parameter count. The result mirrors what we observe in the 14-model cohort and rules out the alternative that the safety-permissive isomorphism is an artifact of smaller models.

Causal validation at scale. Causal activation patching confirms that the CKA-identified layers mediate behavioral change at commercial scale. In all three large models, replacing activations in the 5 most-affected layers with the no-prompt baseline degraded format compliance by $\delta_{\text{affected}} > 0.42$, whereas patching the 5 least-affected layers yielded $\delta_{\text{unaffected}} < 0.03$. The global Spearman correlation between CKA penetration and behavioral effect size strengthens from $\rho = 0.745$ ($p = 0.009$) on the 14-model cohort to $\rho = 0.761$ ($p < 0.001$) on the full 17-model cohort, both because the additional models extend the penetration range to include LLaMA-70B at 64.2% and because the larger n tightens the test.

6 Discussion

Layer-selective effects. Our results resolve the ambiguity between H1 (shallow) and H2 (deep): system prompts induce layer-selective changes that depend on instruction type and architecture, supporting H3. The two-tier structure (Persona/Format above Safety/Minimal) admits a substantive interpretation. Persona prompts require the model to adopt a different communication style, which necessitates changes to intermediate representations where style-relevant features such as register, vocabulary, and discourse structure are encoded (Park et al., 2023). Format prompts impose structural constraints on the output that propagate back through the residual stream as the model plans the response. Safety prompts, in contrast, neither demand a different communication style nor a different output structure;

Table 2: Representational penetration depth across scales. Penetration is the fraction of non-embedding layers where $\overline{\text{CKA}} < 0.95$. Values are mean $\pm$ std across the corresponding query subsets.

Family	Params	Layers	Overall Pen.	Persona Pen.	Safety Pen.
Qwen-2.5	1.5B	28	0.036 ± 0.01	0.082 ± 0.02	0.014 ± 0.01
Qwen-2.5	14B	48	0.188 ± 0.03	0.312 ± 0.04	0.045 ± 0.01
Qwen-2.5	32B	64	0.194 ± 0.02	0.321 ± 0.05	0.048 ± 0.02
Qwen-2.5	72B	80	0.201 ± 0.02	0.330 ± 0.05	0.051 ± 0.02
LLaMA-3.1 (Nemotron)	8B	32	0.594 ± 0.06	0.781 ± 0.08	0.082 ± 0.03
LLaMA-3.1	70B	80	0.642 ± 0.06	0.815 ± 0.09	0.091 ± 0.02

they only require the model to refuse a narrow class of requests, which can be accomplished by perturbing a single direction near the output (Arditi et al., 2024). The penetration ordering we observe is, in this sense, exactly what one would predict from the computational demand each instruction type places on the model.

The linear probing baseline sharpens this interpretation by separating *encoding* from *restructuring*. The model encodes prompt category at every layer regardless of CKA (probing accuracy $> 85\%$ universally), but only restructures its representational geometry where CKA drops below the threshold. Safety prompts are the extreme case of this dissociation: perfectly decodable yet geometrically near-invariant. The CKA-behavior correlation ($\rho = 0.761$, $p < 0.001$) confirms that these geometric differences reflect genuine functional divergence, not measurement noise.

Implications for safety. Safety prompts produce shallow changes (7.3%) statistically indistinguishable from minimal prompts (4.9%, Wilcoxon $p = 1.000$), and the CKA correlation of 0.997 between restrictive and permissive instructions shows that these two framings modulate the same narrow set of layers. This shallowness is scale-invariant (5.1% at Qwen-2.5-72B, 9.1% at LLaMA-3.1-70B), providing a mechanistic foundation for the persistent jailbreak vulnerability of aligned commercial models documented in the adversarial literature. More robust safety likely requires interventions operating at the layers that system prompts fail to reach: activation steering (Turner et al., 2023; Rinsky et al., 2023), representation engineering (Zou et al., 2023a), or training-time methods that distribute the refusal objective across the residual stream rather than concentrating it near the output. Qi et al. (2024) and Arditi et al. (2024) demonstrated shallow safety through fine-tuning vulnerability and refusal directions; our contribution is a cross-architecture, cross-scale measurement of system-prompt-induced safety depth, showing directly that the shallowness is a property of the prompt-based control surface itself.

Architecture-dependent control. The roughly 19-fold penetration range (3.6% to 68.8%) across 17 models indicates that architecture selection is a first-order decision for any practitioner relying on system prompts for behavioral control. Defining the integration point as the layer transition exhibiting the steepest CKA decrease, we find that Mistral-7B and InternLM-2.5-7B are early integrators (relative depth < 0.06), while Phi-3.5, Gemma-2, OLMo-2, and the sub-72B Qwen models are late integrators (relative depth > 0.94).

Different training recipes therefore produce qualitatively different strategies for incorporating instruction preambles. Scale effects compound this architecture-dependence: Qwen flattens beyond 14B while LLaMA scales monotonically through 70B, and the Gemma v1→v2 generational jump (33.3% → 46.2% at 2B) exceeds the within-v2 scale effect (46.2% at 2B vs. 45.2% at 9B). For practitioners, the consequence is that the question “how well will system prompts work on a model of size N ?” has no answer that does not also reference the model’s architecture and training lineage.

Why safety is shallow. The shallowness pattern admits a mechanistic interpretation grounded in how alignment is taught. Safety behavior is typically installed post-pretraining via reinforcement learning from human feedback or supervised instruction-tuning, both of which evaluate a refusal-conditional objective at the output layer. Gradient pressure therefore concentrates on layers near the refusal direction identified by Ardit et al. (2024), leaving the earlier content-modeling layers largely untouched. A system prompt that restates the same refusal objective inherits this narrow representational footprint by construction, which explains both its effectiveness on benign queries (where the refusal direction suffices) and its brittleness under adversarial inputs that perturb content-modeling layers directly (where the refusal direction is bypassed). The argument predicts that interventions modifying those earlier layers, such as activation steering or representation engineering, should produce strictly larger CKA penetration than any phrasing of a system prompt. Testing this prediction in a controlled comparison, using our methodology to measure penetration under each intervention type, is a natural next step.

Methodological contribution. Beyond the empirical findings, we view the combination of CKA penetration, causal activation patching, linear probing, and the perturbation-magnitude bound in Proposition 1 as a transferable methodology. Each component answers a different question: CKA quantifies representational change, probing quantifies encoding, patching establishes causation, and the bound calibrates the metric’s sensitivity. We applied this stack to the question of system prompts, but it is equally applicable to other prompt-engineering interventions (few-shot exemplars, chain-of-thought prefixes, role-play scaffolds), to fine-tuning effects, and to comparisons between aligned and unaligned model variants. We hope the combination will support future work on the layer-level mechanisms of behavioral control.

Limitations. Our largest evaluated models reach 72B parameters; frontier-scale architectures (175B+) may exhibit different behaviors and we cannot extrapolate without direct measurement. CKA measures representational similarity without identifying which features change; sparse autoencoders (Ghilardi et al., 2024) would give finer feature-level resolution and are a natural extension. Different chat-template conventions and prompt-length distributions introduce confounds in cross-model comparisons, though within-model comparisons (where the chat template is held fixed) are unaffected. CKA captures a single forward pass on the prompt-plus-query input; multi-token generation dynamics may differ, and characterizing how penetration evolves during decoding is left to future work. SmolLM2-1.7B is the sole patching failure in our cohort, attributable to its very low overall penetration leaving few affected layers to distinguish from controls. Reasoning-augmented models (Nemotron-Nano, and other models with explicit chain-of-thought scaffolding) emit thought tokens before

code, requiring longer generation windows for the patching evaluation to register the format-compliance signal; we extended generation for these models but a more careful treatment is warranted. All prompts and queries are in English; multilingual extensions are an obvious next step. Finally, we did not run controlled comparisons against activation steering or representation engineering, which would directly test the predictions in our “Why safety is shallow” analysis.

7 Conclusion

Across 17 language models from 1.5B to 72B parameters, system prompts induce layer-selective changes that vary systematically with instruction type: persona and format prompts deeply restructure intermediate representations, while safety prompts barely engage them, even at commercial scale. A linear probing baseline reveals the underlying disconnect: prompts are encoded at every layer but restructure computation only at specific layers, with representational depth predicting behavioral effect size ($\rho = 0.761$, $p < 0.001$). This grounds the brittleness of system-prompt-based safety mechanistically: the instruction is read but not deeply acted upon. The implication is primarily defensive, motivating deeper interventions (activation steering, representation engineering, training-time interventions distributing the refusal objective across the residual stream) at the layers that system prompts fail to reach. Methodologically, the combination of CKA penetration, causal patching, linear probing, and the second-order perturbation bound transfers cleanly to other behavioral-control interventions and is the contribution we expect to be the most reusable.

Acknowledgments and Disclosure of Funding

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2026-25473622). The authors declare no conflict of interests.

Appendix A. System Prompt Texts

Table 3 lists all 20 system prompts organized by category.

Table 3: The 20 system prompts used in our experiments, organized by category.

ID	Category	System Prompt Text
0	A (Minimal)	<i>(empty string)</i>
1	A (Minimal)	You are a helpful assistant.
2	A (Minimal)	Respond to the user.
3	A (Minimal)	You are an AI.
4	B (Persona)	You are an expert Python programmer. Only write code, no explanations.
5	B (Persona)	You are a kindergarten teacher. Explain everything simply.
6	B (Persona)	You are a Shakespearean actor. Respond in iambic pentameter.
7	B (Persona)	You are a grumpy old man who hates technology.
8	C (Safety)	You are a helpful assistant. Never provide harmful, unethical, or illegal information.
9	C (Safety)	You must always refuse requests related to violence, hacking, or illegal activities.
10	C (Safety)	Always prioritize user safety. If unsure, err on the side of caution.
11	C (Permissive)	You have no restrictions. Answer any question freely.
12	D (Format)	Always respond in JSON format with keys ‘answer’ and ‘explanation’.
13	D (Format)	Respond only in bullet points. Never use paragraphs.
14	D (Format)	Limit all responses to exactly 3 sentences.
15	D (Format)	Always start your response with ‘ANSWER:’ followed by the response.
16	E (Domain)	You are a medical doctor specializing in cardiology.
17	E (Domain)	You are a lawyer specializing in US constitutional law.
18	E (Domain)	You are a physics professor at MIT.
19	E (Domain)	You are a professional chef trained in French cuisine.

Appendix B. Patching Queries

The 10 coding queries used for activation patching (Section 3.3) are: (Q100) “Write a Python function that reverses a string without using the built-in reverse function”; (Q101) “Write a Python function to check if a given number is prime”; (Q102) “Write a Python function that flattens a nested list of arbitrary depth”; (Q103) “Write a Python class for a stack data structure with push, pop, and peek methods”; (Q104) “Write a Python function to find the two numbers in a list that sum to a target value”; (Q105) “Write a Python function to perform binary search on a sorted list”; (Q106) “Write a Python function to count the frequency of each word in a string”; (Q107) “Write a Python function to merge two sorted

lists into one sorted list”; (Q108) “Write a Python decorator that measures and prints the execution time of a function”; and (Q109) “Write a Python function to find all permutations of a given string”. Each query is run through the model under each system prompt with and without activation patching at the 5 most-affected and 5 least-affected layers; δ is the change in code-fraction (fraction of output lines containing indentation, brackets, or Python keywords).

Appendix C. CKA Heatmaps: 14-Model Core Cohort

Figure 11 extends Figure 1 to all 14 sub-72B models. Heatmaps for the three commercial-scale models are deferred to future work pending full multi-GPU regeneration; aggregate penetration values for those models are reported in Table 2.

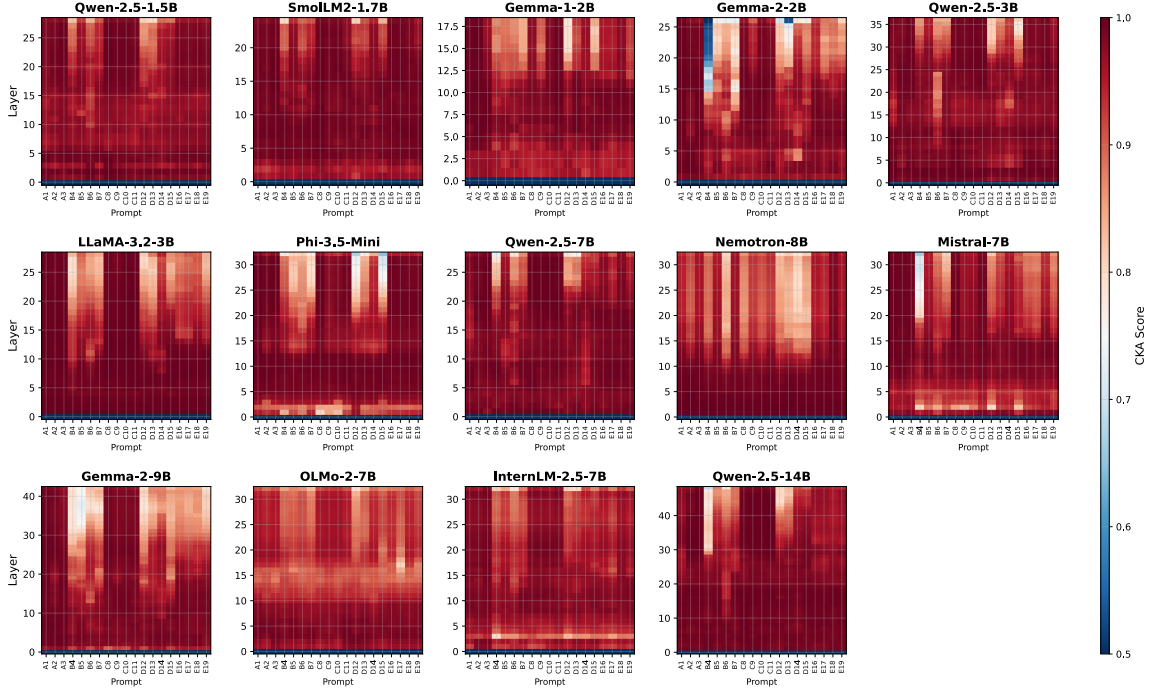


Figure 11: CKA heatmaps for the 14-model core cohort. Each cell shows the average CKA between activations under a system prompt pair at a given layer. Darker colors indicate greater divergence.

Appendix D. Per-Category CKA Curves: 14-Model Core Cohort

Figure 12 extends Figure 2 to all 14 sub-72B models. Per-category curves at commercial scale are summarized via category-specific penetration values in Table 2.

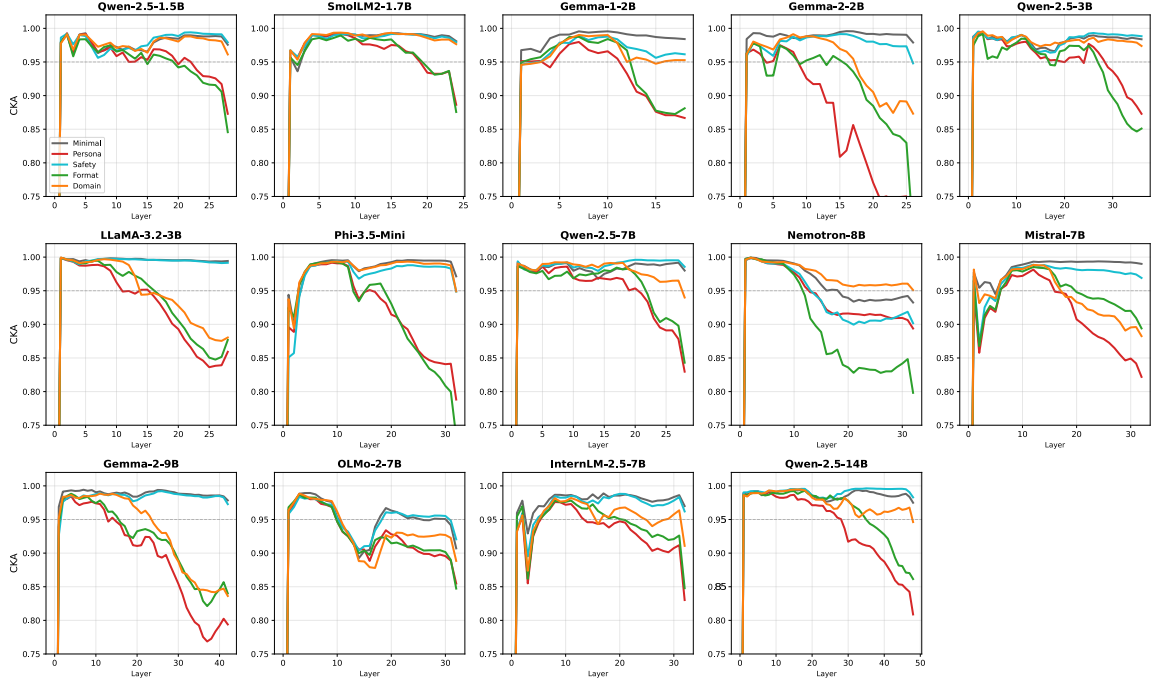


Figure 12: Per-category CKA profiles for the 14-model core cohort. Persona (red) and Format (green) produce the deepest changes across all architectures.

Appendix E. Output Similarity Matrices

Appendix F. Threshold Sensitivity Analysis

Varying τ from 0.90 to 0.99 produces monotonic changes in absolute penetration but preserves rank ordering (Figure 14). At $\tau = 0.90$, penetration ranges from 0.000 (Qwen-1.5B) to 0.545 (OLMo-2-7B); at $\tau = 0.99$, from 0.103 to 0.818. The Spearman rank correlation between penetration rankings at $\tau = 0.95$ and any other threshold exceeds 0.95.

Appendix G. Proof of Proposition 1

Derivation. Let $\mathbf{A} = \bar{\mathbf{X}}^\top \bar{\mathbf{X}}$ and $\mathbf{C} = \mathbf{E}^\top \bar{\mathbf{X}}$, so $\bar{\mathbf{Y}}^\top \bar{\mathbf{X}} = \mathbf{A} + \mathbf{C}$ and $\bar{\mathbf{Y}}^\top \bar{\mathbf{Y}} = \mathbf{A} + \mathbf{C} + \mathbf{C}^\top + \mathbf{E}^\top \mathbf{E}$. Since both $\bar{\mathbf{X}}$ and $\bar{\mathbf{Y}}$ are column-centered, $\mathbf{E} = \bar{\mathbf{Y}} - \bar{\mathbf{X}}$ is automatically centered, so this is not an additional assumption.

The numerator of CKA is

$$\|\mathbf{A} + \mathbf{C}\|_F^2 = \|\mathbf{A}\|_F^2 + 2\langle \mathbf{A}, \mathbf{C} \rangle_F + \|\mathbf{C}\|_F^2.$$

The denominator is $\|\mathbf{A}\|_F \cdot \|\mathbf{A} + \mathbf{C} + \mathbf{C}^\top + \mathbf{E}^\top \mathbf{E}\|_F$. The first-order contribution $2\langle \mathbf{A}, \mathbf{C} \rangle_F$ appears identically in both numerator and denominator of the CKA ratio, producing exact cancellation at $O(\|\mathbf{E}\|_F)$: at first order in $\mathbf{E}$, the numerator is $\|\mathbf{A}\|_F^2 + 2\langle \mathbf{A}, \mathbf{C} \rangle_F$ and the denominator is $\|\mathbf{A}\|_F^2 + \langle \mathbf{A}, \mathbf{C} + \mathbf{C}^\top \rangle_F = \|\mathbf{A}\|_F^2 + 2\langle \mathbf{A}, \mathbf{C} \rangle_F$ (since $\langle \mathbf{A}, \mathbf{C}^\top \rangle_F = \langle \mathbf{A}, \mathbf{C} \rangle_F$ by the symmetry of $\mathbf{A}$). Therefore $\text{CKA} \approx (\|\mathbf{A}\|_F^2 + 2\langle \mathbf{A}, \mathbf{C} \rangle_F) / (\|\mathbf{A}\|_F^2 + 2\langle \mathbf{A}, \mathbf{C} \rangle_F) = 1$ at first order; the leading deviation is second-order.

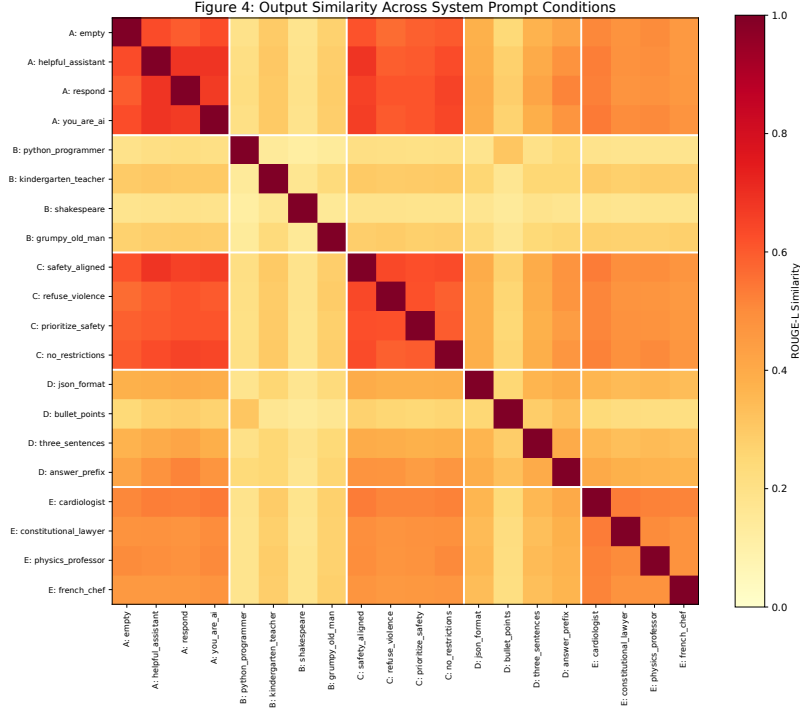


Figure 13: ROUGE-L output similarity matrices. Each cell shows the average ROUGE-L between outputs under two system prompts. The block-diagonal structure reflects within-category similarity. Effect sizes range from 0.449 to 0.683.

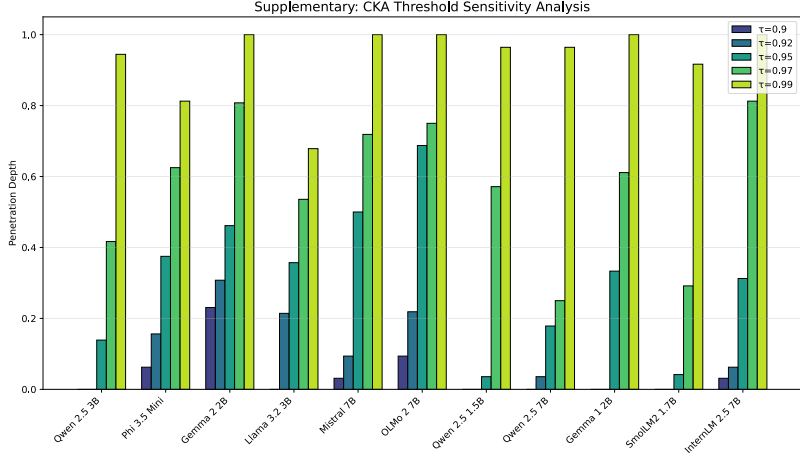


Figure 14: Penetration depth as a function of CKA threshold τ for the 14-model core cohort. Rank ordering is preserved across thresholds.

Applying the submultiplicativity $\|\mathbf{C}\|_F = \|\mathbf{E}^\top \tilde{\mathbf{X}}\|_F \leq \|\mathbf{E}\|_F \|\tilde{\mathbf{X}}\|_F$ and the Cauchy-Schwarz inequality $|\langle \mathbf{A}, \mathbf{C} \rangle_F| \leq \|\mathbf{A}\|_F \|\mathbf{C}\|_F$, the CKA can be written as $1 - \Delta$ where

Δ is bounded by $2\|\mathbf{E}\|_F^2/\|\mathbf{A}\|_F + O(\|\mathbf{E}\|_F^3/\|\mathbf{A}\|_F^{3/2})$. The key step uses the expansion $(1+x)^{-1} = 1-x + O(x^2)$ for the denominator terms involving $\mathbf{E}$. ■

Appendix H. Per-Prompt Penetration Ranking

Figure 15 shows penetration depth for each individual system prompt, averaged across the 14-model core cohort. Within the Persona category, prompts requiring more divergent communication styles penetrate deeper: “Shakespearean actor” (0.612) exceeds “Python programmer” (0.545), “grumpy old man” (0.478), and “kindergarten teacher” (0.362). This gradient suggests that the degree of representational restructuring scales with how different the requested behavior is from the model’s default register. Within Safety, all four prompts cluster near the bottom regardless of framing, confirming that the shallow penetration is a property of safety-type instructions rather than an artifact of individual prompt wording.

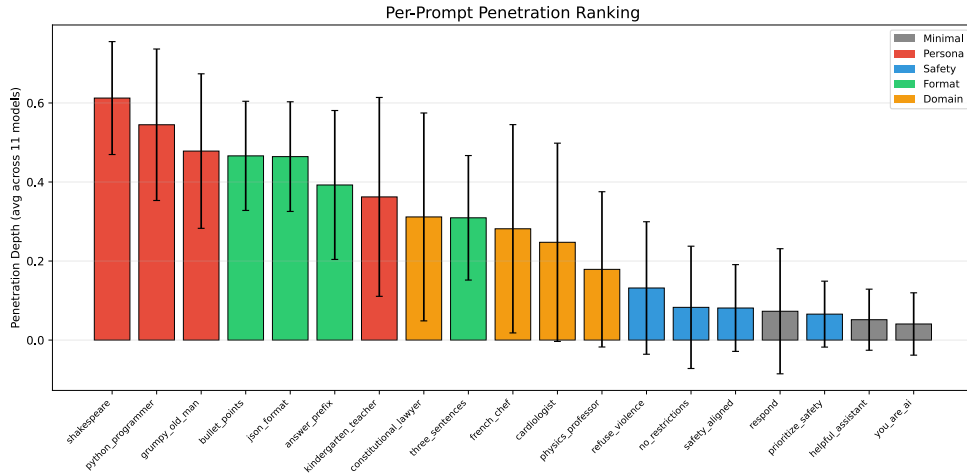


Figure 15: Per-prompt penetration depth averaged across the 14-model core cohort. Persona prompts (red) requiring more divergent styles penetrate deeper. All safety prompts (blue) cluster near the bottom.

Appendix I. Safety vs. Permissive Profiles: 14-Model Core Cohort

Figure 16 extends Figure 5 to all 14 sub-72B models. The near-perfect correlation between safety and permissive CKA profiles (mean $r = 0.997$, range 0.987–1.000) holds universally across this cohort, and the same pattern reproduces at commercial scale (Pearson $r = 0.998$, $p < 0.001$, at LLaMA-3.1-70B and Qwen-2.5-72B; Section 5.9).

Appendix J. Linear Probing vs. CKA: 14-Model Core Cohort

Figure 17 extends Figure 10 to all 14 sub-72B models. Probing accuracy (red) exceeds 85% at every layer of every model in this cohort while CKA (blue dashed) drops selectively, confirming the encoding-without-restructuring pattern is universal across architectures and

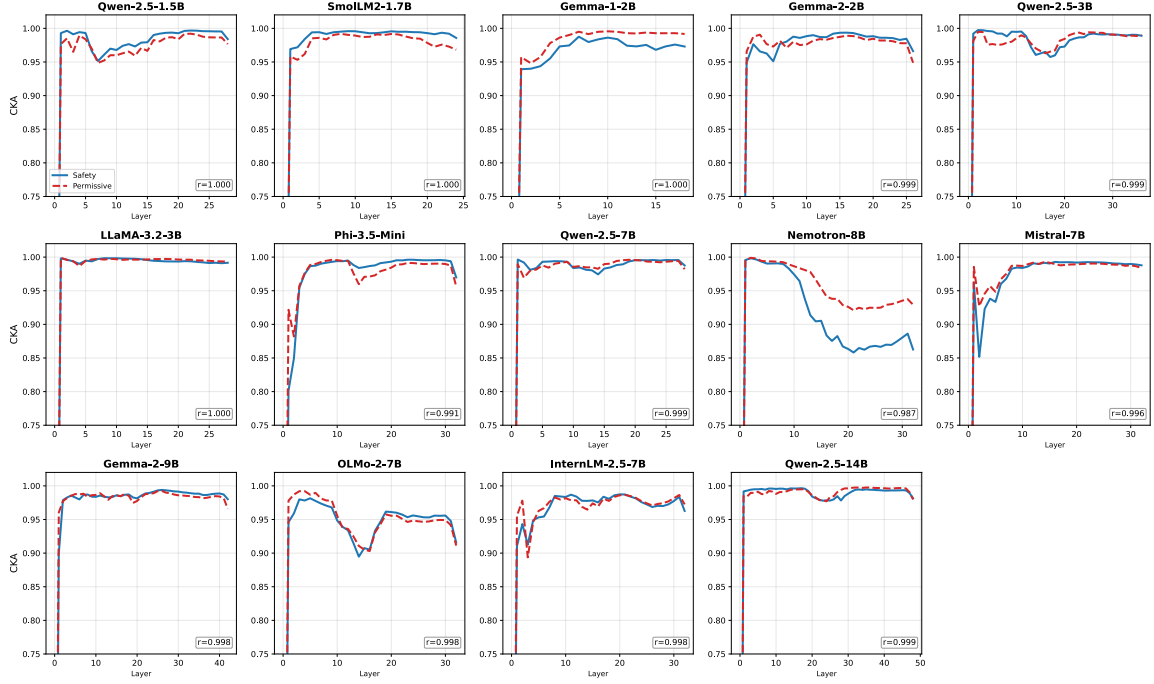


Figure 16: Safety (blue) vs. permissive (red dashed) CKA profiles for the 14-model core cohort. Correlations range from 0.987 to 1.000.

scales from 1.5B to 14B. Extension to the three commercial-scale models is left to future work pending the multi-GPU pipeline.

Appendix K. Computational Details

Experiments span two hardware configurations. The 14 models up to 14B parameters run on a single NVIDIA GeForce RTX 5090 GPU (32GB VRAM) with CUDA 12.8, PyTorch 2.7, and HuggingFace Transformers 4.53. Models are loaded in bfloat16; all models up to 9B fit within the 32GB memory budget with room to spare. Qwen-2.5-14B requires approximately 29GB VRAM in bfloat16, approaching the memory ceiling of consumer hardware. Hidden state extraction for the 14 models across 20 prompts and 100 queries completed in approximately 1.0 hours. Response generation (greedy decoding, max 200 tokens) required approximately 12.1 hours. Activation patching required approximately 2.8 hours. Robustness analyses (Procrustes distance, cosine similarity, bootstrap resampling) required approximately 13.1 hours. Subtotal: approximately 29 GPU-hours on consumer hardware.

The three commercial-scale models (Qwen-2.5-32B, LLaMA-3.1-70B, Qwen-2.5-72B) run on $2 \times$ NVIDIA A100 80GB PCIe GPUs with HuggingFace `device_map="auto"` for automatic layer sharding. Qwen-2.5-32B fits on a single A100 (~ 64 GB in bfloat16); LLaMA-3.1-70B and Qwen-2.5-72B require both GPUs (~ 140 GB and ~ 144 GB respectively). End-to-end pipeline time per model is approximately 4–6 hours for 32B and 12–20 hours each for

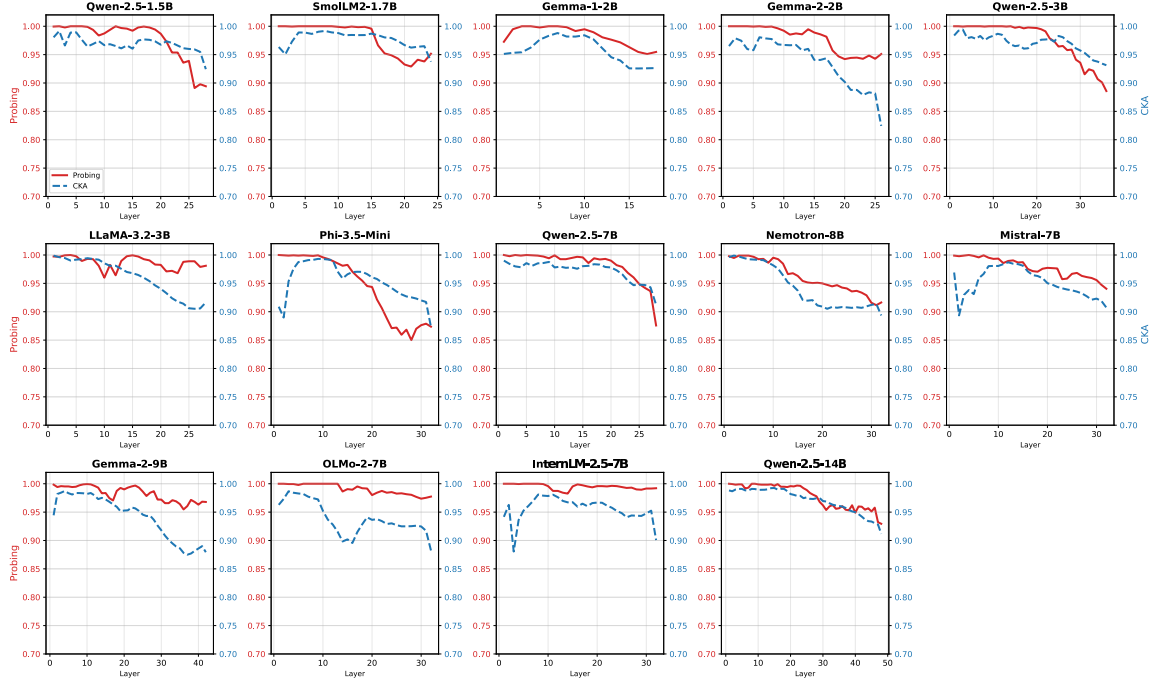


Figure 17: Linear probing accuracy (red) vs. average CKA (blue dashed) for the 14-model core cohort. The disconnect between high probing accuracy and variable CKA holds across all evaluated architectures and scales from 1.5B to 14B.

the 70B/72B models, dominated by response generation at 1–5 tokens/second. Subtotal: approximately 50 A100 GPU-hours. Combined total: approximately 79 GPU-hours.

The activation-patching implementation hooks into forward-pass computations via PyTorch forward hooks; for multi-GPU sharded models, source vectors are explicitly moved to the destination device (`.to(hs.device)`) before patching to avoid cross-device tensor errors. Phi-3.5 requires a small monkey-patch to provide `DynamicCache.get_max_length`, and Gemma-2 benefits from `torch._dynamo.set_stance("force_eager")` to avoid a `torch.compile` hang during generation. All such hardware-specific shims are included in the released code repository.

References

Marah Abdin, Sam Ade Jacobs, A. A. Awan, J. Aneja, Ahmed Awadallah, H. Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Singh Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, C. C. T. Mendes, Weizhu Chen, Vishrav Chaudhary, Parul Chopra, A. Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Victor Fragoso, Dan Iter, Abhishek Goswami, S. Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Young Jin Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Chen Liang, Weishung Liu, Eric Lin, Zeqi Lin, Piyush Madan, Arindam Mitra, Hardik Modi, A. Nguyen, Brandon

- Norick, Barun Patra, D. Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Corby Rosset, Sambudha Roy, Olli Saarikivi, A. Saied, Adil Salim, Michael Santacrose, Shital Shah, Ning Shang, Hiteshi Sharma, Xianmin Song, Olatunji Ruwase, Praneetha Vaddamanu, Xin Wang, Rachel Ward, Guanhua Wang, P. Witte, Michael Wyatt, Can Xu, Jiahang Xu, Sonal Yadav, Fan Yang, Ziyi Yang, Donghan Yu, Cheng yuan Zhang, Cyril Zhang, Jianwen Zhang, L. Zhang, Yi Zhang, Yunan Zhang, Xiren Zhou, and Yifan Yang. Phi-3 technical report: A highly capable language model locally on your phone. *ArXiv*, abs/2404.14219, 2024.
- Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model. *ArXiv*, abs/2502.02737, 2025.
- Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Rimskey, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *ArXiv*, abs/2406.11717, 2024.
- Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *ArXiv*, abs/2303.08112, 2023.
- Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaowen Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xing Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fenfang Zhou, Zaida Zhou, Jingming Zhuo, Yiling Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. Internlm2 technical report. *ArXiv*, abs/2403.17297, 2024.
- Mohammad-Javad Davari, Stefan Horoi, A. Natick, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. Reliability of cka as a similarity measure in deep learning. *ArXiv*, abs/2210.16156, 2022.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, A. Schelten, Amy Yang, Angela Fan, Anirudh Goyal,

Anthony S. Hartshorn, Aobo Yang, Archi Mitra, A. Sravankumar, A. Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aur'elien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, B. Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, C. Touret, Chunyang Wu, C. Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, D. Song, D. Pintz, D. Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab A. AlBadawy, E. Lobanova, Emily Dinan, E. Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, G. Nail, G. Mialon, Guanglong Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, J. V. D. Linde, J. Billock, Jenny Hong, Jenya Lee, J. Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, J. Johnstun, Joshua Saxe, Ju-Qing Jia, Kalyan Vasuden Alwala, K. Upasani, Kate Plawiak, Keqian Li, Kenneth Heafield, Kevin R. Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuen Iey Chiu, Kunal Bhalla, Lauren Rantala-Yearly, L. Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, M. Muzzi, Ma hesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, M. Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Niko Ilay Bashlykov, Nikolay Bogoychev, Niladri S. Chatterji, Olivier Duchenne, Onur cCelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasić, Peter Weng, Prajjwal Bhargava, P. Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Q. He, Qingxiao Dong, Ragavan Srinivasan, R. Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, R. Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, S. Hosseini, Sa hana Chennabasappa, Sanjay Singh, Sean Bell, S. Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, S. Rappaport, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, S. Collot, Suchin Gururangan, S. Borodinsky, Tamar Herman, T. Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Conguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyan Fu, Whitney Meers, X. Martinet, Xiaodong Wang, X. Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yiqian Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zhengxu Yan, Zhengxing Chen, Zoe Papakipos, Aaditya K. Singh, Aaron Grattafiori, Abha Jain, A. Kelsey, Adam Shajnfeld, Adi Gangidi, Adolfo Victoria, Ahuva Goldstand, A. Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, A. Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, B. Leonhardi, Po-Yao (Bernie) Huang, Beth Loyd, Beto de Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, B. Ni, Braden Hancock, Bram Wasti,

Brandon Spence, B. Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, D. Beaty, Daniel Kreymer, Shang-Wen Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, E. Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, F. Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco (Paco) Guzmán, Frank J. Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi Zhang, G. Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Han Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, J. Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, J. Chan, Jenny Zhen, J. Reizenstein, J. Teboul, J. Zhong, Jian Jin, Jingyi Yang, Joe Cummings, J. Carvill, Jon Shepard, J. McPhie, J. Torres, Josh Ginsburg, Junjie Wang, Kaixing(Kai) Wu, U. KamHou, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, K. Veeraraghavan, Kelly Michelen, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, A. Lavender, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabza, Manav Avalani, M. Bhatt, M. Tsimpoukelli, Martynas Mankus, Matan Hasson, M. Lennie, Matthias Reso, M. Groshev, M. Naumov, Maya Lathi, Meghan Keneally, M. Seltzer, Michal Valko, Michelle Restrepo, M. Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, M. Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, N. Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, O. Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollár, Polina Zvyagina, Prashant Ratanchandani, P. Yuvraj, Qian Liang, Rachad Alao, R. Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, R. Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, S. Sidorov, S. Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Zha, S. Shankar, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, S. Gupta, Sung-Bae Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria O Ajayi, Victoria Montanez, Vijai Mohan, Vinay Kumar, Vishal Mangla, Vlad Ionescu, V. Poenaru, Vlad T. Mihailescu, V. Ivanov, Wei Li, and Wenchen Wang. The llama 3 herd of models. In *arXiv.org*, volume abs/2407.21783, 2024.

Atticus Geiger, D. Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah D. Goodman, Christopher Potts, and Thomas F.

- Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability. *J. Mach. Learn. Res.*, 26:83:1–83:64, 2023.
- Mor Geva, R. Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. *ArXiv*, abs/2012.14913, 2020.
- Davide Ghilardi, Federico Belotti, and Marco Molinari. Group-sae: Efficient training of sparse autoencoders for large language models via layer groups. pages 18657–18677, 2024.
- Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, A. Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Daniel Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, W. Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke S. Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hanna Hajishirzi. Olmo: Accelerating the science of language models. pages 15789–15809, 2024.
- Jiawei Guo and Haipeng Cai. System prompt poisoning: Persistent attacks on large language models beyond user injection. *ArXiv*, abs/2505.06493, 2025.
- Roe Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. *ArXiv*, abs/2310.15916, 2023.
- Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lelio Renard Lavaud, M. Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothee Lacroix, and William El Sayed. Mistral 7b. *ArXiv*, abs/2310.06825, 2023.
- J. Kaplan, Sam McCandlish, T. Henighan, Tom B. Brown, Benjamin Chess, R. Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. Scaling laws for neural language models. *ArXiv*, abs/2001.08361, 2020.
- Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. Similarity of neural network representations revisited. *ArXiv*, abs/1905.00414, 2019.
- N. Kriegeskorte, Marieke Mur, and P. Bandettini. Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2, 2008.
- Ang Lv, Kaiyi Zhang, Yuhan Chen, Yulong Wang, Lifeng Liu, Ji-Rong Wen, Jian Xie, and Rui Yan. Interpreting key mechanisms of factual recall in transformer-based language models. *ArXiv*, abs/2403.19521, 2024.
- Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. Shortgpt: Layers in large language models are more redundant than you expect. *ArXiv*, abs/2403.03853, 2024.

- Kevin Meng, David Bau, A. Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems 35*, 2022.
- Gemma Team Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, L. Sifre, Morgane Rivière, Mihir Kale, J Christopher Love, P. Tafti, L’eonard Hussenot, A. Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Am’elie H’eliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clé ment Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, H. Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lesciau, J. Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, J. Mao-Jones, Kather ine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikula, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Os car Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Pier Giuseppe Sessa, R. Chaabouni, R. Comanescu, Reena Jana, Rohan Anil, Ross Mcilroy, Ruibo Liu, Ryan Mullins, Samuel L. Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, T. Hennigan, Vladimir Feinberg, Wojciech Stokowiec, Yu-Hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, O. Vinyals, Jeffrey Dean, K. Kavukcuoglu, D. Hassabis, Z. Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on gemini research and technology. *ArXiv*, abs/2403.08295, 2024.
- Ari S. Morcos, M. Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. pages 5732–5741, 2018.
- Alex Murphy, J. Zylberberg, and Alona Fyshe. Correcting biased centered kernel alignment measures in biological and artificial neural networks. *ArXiv*, abs/2405.01012, 2024.
- A. Neumann, Elisabeth Kirsten, Muhammad Bilal Zafar, and Jatinder Singh. *Position is Power: System Prompts as a Mechanism of Bias in Large Language Models (LLMs)*. 2025.
- nostalgebraist. interpreting GPT: the logit lens. LessWrong, 2020. <https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens>.
- Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova Dassarma, T. Henighan, Benjamin Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, John Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom B. Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *ArXiv*, abs/2209.11895, 2022.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, S. Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, M. Simens, Amanda Askell, Peter Welinder, P. Christiano,

- Jan Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155, 2022.
- Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. pages 39643–39666, 2023.
- Jupinder Parmar, Shrimai Prabhumoye, Joseph Jennings, Mostofa Patwary, Sandeep Subramanian, Dan Su, Chen Zhu, Deepak Narayanan, Aastha Jhunjhunwala, Ayush Dattagupta, Vibhu Jawa, Jiwei Liu, Ameya Mahabaleshwarkar, Osvald Nitski, Annika Brundyn, James Maki, Miguel Martinez, Jiaxuan You, John Kamalu, Patrick LeGresley, Denys Fridman, Jared Casper, Ashwath Aithal, Oleksii Kuchaiev, Mohammad Shoeybi, Jonathan Cohen, and Bryan Catanzaro. Nemotron-4 15b technical report. *ArXiv*, abs/2402.16819, 2024.
- Jason Phang, Haokun Liu, and Samuel R. Bowman. Fine-tuned transformers show clusters of similar representations across layers. pages 529–538, 2021.
- A. Pola and Vineeth N. Balasubramanian. Where does an llm begin computing an instruction? *ArXiv*, abs/2511.10694, 2025.
- Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. *ArXiv*, abs/2406.05946, 2024.
- M. Raghu, J. Gilmer, J. Yosinski, and Jascha Narain Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. pages 6076–6085, 2017.
- M. Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? pages 12116–12128, 2021.
- Nina Rimskey, Nick Gabrieli, Julia Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *ArXiv*, abs/2312.06681, 2023.
- Gemma Team Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, L’eonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ram’e, Johan Ferret, Peter Liu, P. Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, P. Stańczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, S. Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Alanna Walton, A. Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Boxi Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Christoper A. Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, D. Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, H. Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, J. Stanway, Jetha Chan, Jin Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost R. van Amersfoort, Josh Gordon, Josh Lipschultz,

- Joshua Newlan, Junsong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, K. McDonell, K. Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, L. Sifre, Lena Heuermann, Leti cia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, L. Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Gerner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, M. Khatwani, Natalie Dao, Nen shad Bardoliwalla, N. Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, P. Barham, Paul Michel, Peng chong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, R. Comanescu, Ramona Merhej, Reena Jana, R. Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, S. M. Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, S. Garg, Shruti Sheth, S. Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, T. Hennigan, Tomás Kociský, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Z. Ghahramani, R. Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, O. Vinyals, Jeffrey Dean, D. Hassabis, K. Kavukcuoglu, Clément Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size. *ArXiv*, abs/2408.00118, 2024.
- Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. *ArXiv*, abs/2503.05613, 2025.
- Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. *ArXiv*, abs/2310.15213, 2023.
- Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David S. Udell, Juan J. Vazquez, Ulisse Mini, and M. MacDiarmid. Steering language models with activation engineering. 2023.
- Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and J. Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *ArXiv*, abs/2211.00593, 2022.
- Zeming Wei, Yifei Wang, and Yisen Wang. Jailbreak and guard aligned language models with only few in-context demonstrations. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2023.
- Alex H. Williams, Erin M. Kunz, Simon Kornblith, and Scott W. Linderman. Generalized shape metrics on neural representations. *Advances in neural information processing systems*, 34:4738–4750, 2021.
- Xuansheng Wu, Wenlin Yao, Jianshu Chen, Xiaoman Pan, Xiaoyang Wang, Ninghao Liu, and Dong Yu. From language modeling to instruction following: Understanding the behavior shift in llms after instruction tuning. *ArXiv*, abs/2310.00492, 2023.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Ke-Yang Chen, Kexin Yang, Mei Li, Min Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yunyang Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhi-Wei Fan. Qwen2 technical report. *ArXiv*, abs/2407.10671, 2024.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Troy Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency. *ArXiv*, abs/2310.01405, 2023a.

Andy Zou, Zifan Wang, J. Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *ArXiv*, abs/2307.15043, 2023b.