MAXBENCH: A MULTINOMIAL CONCEPT RECOVERY BENCHMARK

Anonymous authors
Paper under double-blind review

ABSTRACT

Concept localization and activation steering are increasingly common in interpretability research and its applications. Many existing methods implicitly assume binary or one-dimensional concepts, but many concepts are not binary: some contain many values, hierarchical structure, ordered values, among others. Recent work has explored multinomial concepts and found that their representation geometries are far more varied than those of binary concepts; this makes it unclear which methods or assumptions, and thus which localization and steering methods, are most appropriate. In this study, we introduce MAXBENCH, a geometry-agnostic evaluation framework for multinomial concept representations based on sampling from the recovered concept representation. Comparing localization methods across a range of geometry types, concepts, and models, we find the following: (i) where the concept representation is centered often matters as much as (and sometimes more than) the choice of basis and the dimensionality of the concept representation; (ii) affine subspaces generally outperform directions and linear subspaces; (iii) methods within the same geometry class tend to perform similarly; and (iv) no method consistently outperforms prompting, especially for larger models. These findings underscore the importance of expanding meta-evaluations of interpretability research to concepts with more varied internal structure.

1 INTRODUCTION

Among the more influential ideas in interpretability is that some human-understandable concepts are represented as linear directions in activation space. This abstraction has enabled both insights into model internals (Mikolov et al., 2013; Bolukbasi et al., 2016; Elhage et al., 2021; Park et al., 2023; Bricken et al., 2023; Nanda et al., 2023; Huben et al., 2024) as well as techniques for modifying model behavior at inference time. Steering methods (Turner et al., 2024; Li et al., 2023; Zou et al., 2025; Subramani et al., 2022) provide a lightweight and interpretable alternative to prompting and fine-tuning (albeit with somewhat lower performance than prompting (Wu et al., 2025)). For binary concepts such as refusal (Arditi et al., 2024), a steerable representation can often be captured by a single direction in activation space that separates the two states.[1] However, this intuition becomes less straightforward for more complex concepts, which may require richer representations.

Indeed, recent evidence suggests that concepts in language models can take many forms: e.g., they may be polytopes (Park et al., 2025) or may lie on multidimensional (and potentially non-linear) manifolds (Engels et al., 2025; Shafran et al., 2026; Tiblias et al., 2026; Karkada et al., 2026; Bhalla et al., 2026; Wurgaft et al., 2026; Gurnee et al., 2026; Modell et al., 2025; Fel et al., 2026a; Sarfati et al., 2026; Feucht et al., 2026). With regards to evaluation and steering, moving from one-dimensional/binary to multidimensional/multinomial concepts therefore introduces several challenges. First, the space of possible geometries becomes much larger, and may depend on the concept. In the multidimensional setting, a concept may be represented by a set of directions, a Gaussian cloud within the representation space, linear or affine subspaces or non-linear manifolds, among many other possibilities. Second, the dimensionality (or rank) of a concept representation is generally not known in advance and likely varies significantly across concepts. While some concepts have an intuitive

[1] Although see recent work arguing that non-linear manifolds more faithfully capture many concepts (Wurgaft et al., 2026; Fel et al., 2026a; Bhalla et al., 2026).

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

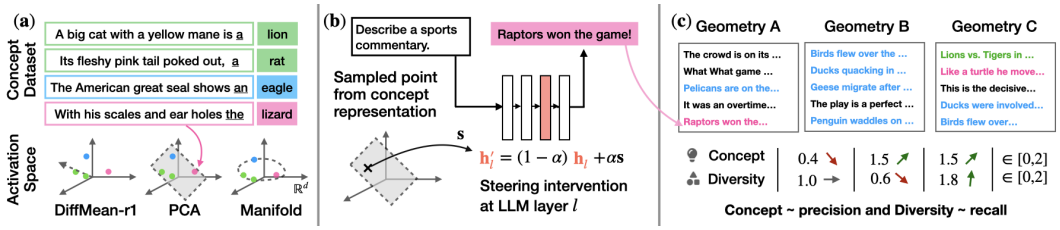


Figure 1: Evaluating multinomial concept representations via the MAXBENCH dataset. **(a)** For a given multinomial concept dataset we can localize a concept geometry in the activation space of a LLM via different methods §3.1. **(b)** Neutral prompts generate concept-dependent continuations when steered via samples from a specific concept §3.2. **(c)** We evaluate the exclusivity and completeness of different concept representations from a set of continuations that were generated for several samples s and neutral prompts. Taken together, several metrics define the final MAXBENCH score §3.3.

geometry (e.g., days of the week tend to form a circle with ordered values; Engels et al., 2025) we cannot assume *a priori* that any given concept will have such a structure.

We therefore ask two questions. First, how can we measure and compare the quality of concept representations in a geometry-agnostic manner? Second, do there exist (classes of) methods or geometries that generally perform well for multinomial concept localization—and if so, what factors explain their performance? To investigate these questions, we first study methods (both existing and novel to this work) for recovering multidimensional concept geometries. We then propose a sampling-based intervention method for steering multinomial concepts in language models. To evaluate this, we introduce MAXBENCH, a framework for evaluating and comparing representations of multinomial concepts. We propose data and metrics to evaluate multinomial concept recovery in terms of precision (capturing only the target concept) and recall (covering the concept fully).

Our main contributions are as follows:

1. We propose a sampling-based steering framework for evaluating how well a given representation captures a target concept. This framework is geometry-agnostic.
2. We introduce MAXBENCH, which applies this framework to 6 concepts and 4 models, and features a novel diversity score that measures how completely steering throughout the space covers the instances and (where applicable) subcategories of a concept.
3. We introduce affine generalizations of existing steering methods for recovering multinomial concept representations.
4. We find that affine representations outperform directions and linear subspaces, and that this is primarily because the offset of a representation often matters more than its dimensionality (and sometimes more than the choice of basis, especially in smaller models). Nonetheless, no method consistently outperforms prompting, especially for larger models.

2 PRELIMINARIES

We informally define a *concept representation* as a geometric object in a model's activation space that captures the boundaries of and variance within a semantic concept, such as ANIMALS or DAYS of the week. Ideally, a concept representation should have high precision (i.e., only information relevant to the concept can be extracted from this representation), and have high recall (i.e., the full range of values of the concept can be recovered from the subspace). For example, if concept C is represented by an affine subspace $\mathbb{A}$, then all information required to recognize or generate text about any value of C (but no more) should be recoverable from $\mathbb{A}$, while directions orthogonal to it should ideally contain no additional information specific to the concept.

Geometries. We consider linear and affine subspaces (including existing one-dimensional methods), as well as non-linear manifolds (Wurgaft et al., 2026; Karkada et al., 2026; Engels et al., 2025; Sarfati et al., 2026). For concision, we focus primarily on affine subspaces in the main text, as this is the most general linear geometry applicable to all concepts considered in our experiments.

Formally, let affine subspace $\mathbb{A} \subset \mathbb{R}^d$ be of the form $\mathbb{A} = \{\boldsymbol{L}\boldsymbol{q} + \boldsymbol{b} \mid \boldsymbol{q} \in \mathbb{R}^k\}$, where $\boldsymbol{L} \in \mathbb{R}^{d \times k}$ is a matrix with orthonormal columns defining a k-dimensional linear subspace, $\boldsymbol{b} \in \mathbb{R}^d$ is an offset

vector, and vector $\boldsymbol{q} \in \mathbb{R}^k$ denotes coefficients on the subspace's bases. A linear subspace is a special case of affine subspaces where the offset $\boldsymbol{b}$ is the origin $\mathbf{0}$.

We additionally study non-linear manifolds. Here, we use non-linear manifold to refer to a manifold (Lee, 2012) that cannot be cast as an affine subspace of $\mathbb{R}^d$. Unlike an affine subspace, which is globally linear, a non-linear manifold may exhibit curvature while remaining locally Euclidean. In this work, we study manifold representations that are obtained by spline-fitting as introduced in Sarfati et al. (2026); Wurgaft et al. (2026). In their current form, these methods can only be applied to concepts with ordered values, such as DAYS.[2]

Data. We assume that the concept geometries described above correspond to low-dimensional structures in the space of language model hidden activations from the residual stream at a given layer. Specifically, for a set of m input tokens $x_{i=1}^m$ that occur just before a concept-related token, we observe hidden activation vectors $\{\boldsymbol{h}^{(i)}\}_{i=1}^m$, where $\boldsymbol{h}^{(i)} \in \mathbb{R}^d$; d is the dimensionality of the language model representations. We assume that each concept C is represented by a low-dimensional structure of dimensionality r, where $r < d$, for example, an affine subspace $\mathbb{A}_C \subset \mathbb{R}^d$ or non-linear manifold $\mathbb{M}_C \subset \mathbb{R}^d$ embedded in this ambient space. We will use "concept recovery" to refer to methods used to locate these geometric objects in language model activations.

Recovering a concept geometry requires observing a concept across diverse contexts. In realistic settings, we may have activations for only a subset of instances belonging to a concept: When learning the concept geometry for ANIMALS, it may be unrealistic to collect a dataset complete with all possible instances and corresponding activations. A good concept recovery method should nevertheless recover a geometry that generalizes well to unseen instances in spite of incomplete data.

Based on these properties, we construct a dataset consisting of six concepts to understand the geometry of semantic representations in language models (Lee et al., 2026; Park et al., 2025): ANIMALS, COUNTRIES, VEHICLES, PLANTS, DAYS, and YEARS. Details of our dataset are given in §4.

3 MAXBENCH: A MULTINOMIAL CONCEPT RECOVERY BENCHMARK

We introduce MAXBENCH, a benchmark for evaluating the precision and coverage of a concept recovery method. The evaluation pipeline consists of three stages (Fig. 1). First, we apply different concept localization methods to recover different concept geometries (§3.1). Second, we introduce a sampling-based evaluation where we sample points from the recovered object and use them to intervene on language model activations (§3.2). Third, we score the resulting outputs using metrics designed to measure the precision and recall over the concept given the recovered geometry (§3.3). While we cannot claim to know the true concept geometry, this framework provides an extrinsic framework for measuring the utility of these methods for faithfully steering multinomial concepts with high coverage.

Table 1: Comparison of concept localization methods in MAXBENCH. **Labels** indicates whether a method is supervised or unsupervised. **Data** describes the type of data used: concept-specific data versus general-purpose web text from The Pile (Gao et al., 2020). **Pairs** indicates whether both positive and negative examples are required. *SAE and MFA are trained without contrastive supervision, but positive and negative examples are used for post-hoc feature selection.

Method	Geometry	Labels	Data	Pairs
DiffMean-r1	Direction	✓	Concept	✓
ReFT-r1	Direction	✓	Concept	✓
DiffMean	Affine subspace	✓	Concept	✗
Probe	Affine subspace	✓	Concept	✗
Schatten Probe	Affine subspace	✓	Concept	✗
PCA	Affine subspace	✗	Concept	✗
Factor Analysis	Affine subspace	✗	Concept	✗
SAE	Linear subspace	✗	Pile	✗*
MFA	Affine subspace	✗	Pile	✗*
Spline Fitting	Non-linear manifold	✓	Concept	✗

3.1 CONCEPT LOCALIZATION METHODS

We consider concept localization methods that differ in three key aspects: the geometry they assume, how the parameters (if any) are set, and supervision type, which determines if the method uses only positive examples (i.e., valid

[2]Extending these methods to unordered categorical concepts like ANIMALS is non-trivial, so we leave evaluation of non-linear manifold methods on unordered concepts to future work.

instances of the concept) or positive–negative pairs (i.e., pairs of inputs where one contains a concept instance and the other does not). See Tab. 1, and §A.1.

The input to each method is a set of activation vectors $\{\boldsymbol{x}^{(i)}\}_{i=1}^{m}$, with $\boldsymbol{x}^{(i)} \in \mathbb{R}^d$. For methods that require supervision, we supply labels $y^{(i)} \in \{0, \ldots, K-1\}$ for each activation. For one-dimensional methods, we use binary labels $y^{(i)} \in \{0, 1\}$, indicating the absence or presence of the concept.

Directions. A direction is a linear one-dimensional object in activation space. Common methods for finding concept-related directions include *Difference in Means* (DiffMean-r1; Marks & Tegmark, 2024), which defines the concept direction as the difference between the mean activations of positive and negative examples defined as $\boldsymbol{d} = \frac{1}{|\{i:y^{(i)}=1\}|}\sum_{i:y^{(i)}=1}\boldsymbol{x}^{(i)} - \frac{1}{|\{i:y^{(i)}=0\}|}\sum_{i:y^{(i)}=0}\boldsymbol{x}^{(i)}$.

Rank-1 Representation Finetuning (ReFT-r1;Wu et al., 2025; 2024) learns a direction $\boldsymbol{d}$ from labeled positive and negative examples, where positive responses express the target concept and negative responses do not. The learned direction is used both to detect the presence of the concept in model representations and to steer the model toward it, with $\boldsymbol{d}$ optimized through a language modeling objective with sparsity regularization. Further details are provided in Wu et al. (2024). For 1D methods we only have direction $\boldsymbol{d}$ and $\boldsymbol{b} = \boldsymbol{0}$.

Unlike the previous methods that recover a single concept direction, *sparse autoencoders* (SAEs; Bricken et al., 2023; Huben et al., 2024) learn a dictionary of feature directions $f_{\text{enc}}(\boldsymbol{x}) = \text{ReLU}(\boldsymbol{L}\boldsymbol{x} + \boldsymbol{b})$. Each dictionary element corresponds to a one-dimensional direction in activation space. We use existing SAEs in our experiments. To identify features corresponding to the target concept, we score each feature by the Matthews Correlation Coefficient (MCC) between its activations and a binary label indicating whether the point is related or unrelated to the concept, and retain the top-k features. Each feature is a one-dimensional direction. We use the top-k SAE feature vectors as the columns of the recovered basis, $\boldsymbol{L} = [\boldsymbol{w}_{f_1} \mid \cdots \mid \boldsymbol{w}_{f_k}]$ and sample from the subspace spanned by these directions. We set the offset to $\boldsymbol{b} = \boldsymbol{0}$. Details in §A.2.

Affine subspaces. Affine concept recovery methods aim to locate a k-dimensional affine subspace $\mathbb{A} = \{\boldsymbol{L}\boldsymbol{q} + \boldsymbol{b} \mid \boldsymbol{q} \in \mathbb{R}^k\}$ rather than a single direction. The affine methods listed in Tab. 1 differ primarily in how $\boldsymbol{L} \in \mathbb{R}^{d\times k}$ is estimated: *PCA* (Pearson, 1901) and *Factor Analysis* (Spearman, 1961) compute an orthonormal $\boldsymbol{L}$ directly. For *Linear Probe* (Alain & Bengio, 2017) and *Schatten Probe* (Braun et al., 2025; Giampouras et al., 2020) we first learn a linear classifier $\mathbf{W} \in \mathbb{R}^{d\times d}$ from the activations (with a p-regularizer). $\boldsymbol{L}$ is then composed from the top singular vectors of the learned matrix.

We extend *Difference in Means* to find affine subspaces (a variant we will call *DiffMean (Pairs)*) as follows: for each pair of subcategories c_i, c_j and their corresponding activations $\mathbb{X}_{c_j}$, we compute the difference vector $\boldsymbol{d}_{i,j} = \boldsymbol{\mu}_{c_i} - \boldsymbol{\mu}_{c_j}$, where $\boldsymbol{\mu}_{c_j} = \frac{1}{|\mathbb{X}_{c_j}|}\sum_{\boldsymbol{x}\in\mathbb{X}_{c_j}}\boldsymbol{x}$. We stack $\boldsymbol{d}_{i,j}$ to form $\boldsymbol{D}$, apply SVD, and define $\boldsymbol{L}$ from its leading singular directions in activation space.

For all affine methods, we set the offset $\boldsymbol{b}$ to the mean of all concept-related activations; we leave other strategies for estimating the offset to future work.

A *Mixture of Factors Analyzer* (MFA; Shafran et al., 2026) learns a collection of low-dimensional Gaussian components, each associated with an affine subspace $\mathbb{A}_c$. Each component c has a centroid $\boldsymbol{b}_c \in \mathbb{R}^d$ and factor-loading matrix $\boldsymbol{L}_c \in \mathbb{R}^{d\times k}$ and MFA models the activation space as $\{\mathbb{A}_c\}_{c=1}^{C}$. For a target concept, we select the component $c^\star$ that is most associated with the target concept and use $\mathbb{A}_{c^\star}$ as the recovered concept representation. The method for selecting $\mathbb{A}_{c^\star}$ is described in §A.3.

Non-linear Manifolds. Following Wurgaft et al. (2026), we first compute a centroid for each concept value and project these centroids into a low-dimensional subspace using PCA. A cubic spline is then fit through the projected centroids; this forms the concept manifold. We use the implementation of Wurgaft et al. (2026) to fit the spline and its associated centroids.

3.2 Sampling-based Evaluation

After obtaining a concept representation, we evaluate whether points contained in that representation correspond to the target concept. Intuitively, if a representation faithfully captures a concept, then steering LM representations toward points sampled from the concept representation should consis-

tently yield instances of that concept; different samples should also cover the full range of instances and subcategories within the concept (rather than a narrow subset of instances/categories).

For a recovered affine subspace $\mathbb{A}_{\text{recovered}} = \{\boldsymbol{L}\boldsymbol{c} + \boldsymbol{b} \mid \boldsymbol{c} \in \mathbb{R}^k\}$, we sample n points $\{\boldsymbol{p}^{(i)}\}_{i=1}^n$ from the subspace by drawing coefficients $\boldsymbol{z}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_k)$ and computing $\boldsymbol{p}^{(i)} = \boldsymbol{L}\boldsymbol{z}^{(i)} + \boldsymbol{b}$. Since affine subspaces are unbounded, we select σ empirically for each model, method, and rank on the validation set to ensure sampled points remain within the typical range of the model's activation space (§B.1). A linear subspace is a special case of affine subspaces with $\boldsymbol{b} = \boldsymbol{0}$, and a direction corresponds to a linear subspace with $r = 1$.

For a recovered non-linear manifold $\mathbb{M}$, we sample n points $\{\boldsymbol{p}^{(i)}\}_{i=1}^n$ by defining a global parameter $s \in [0, K)$, where K is the number of concept values. Each integer k corresponds to the centroid of the kth concept value. For a sampled value $s^{(i)} \sim \mathcal{U}(0, K)$, we write $s^{(i)} = k + \gamma$, where k identifies the corresponding spline segment and $\gamma \in [0, 1)$ specifies the position along the spline segment connecting centroids k and $k + 1$. We then evaluate this spline at $s^{(i)}$ to obtain the sampled point $\boldsymbol{p}^{(i)}$. Repeating this procedure yields samples from the spline-based manifold. If the spline forms a closed loop, as in the case of days of the week, we include the spline segment connecting the final concept-value centroid back to the first.

We intervene on the language model's representations using a sampled point and observe the resulting generated output. Interventions are applied at the same layer from which the concept representations are extracted. For sampled point $\boldsymbol{s}$ and hidden activation $\boldsymbol{h}$, we intervene as follows:

$$\boldsymbol{h}' = (1 - \alpha)\boldsymbol{h} + \alpha \boldsymbol{s}.^{3}$$

The sample determines the location within the concept representation, and α determines the strength of the intervention. We select α using the validation set via a grid search (see §B.1). During generation, we exponentially decay α across decoding steps as $\alpha_t = \alpha_0 \cdot \beta^{-t}$ where $\beta = (\alpha_0/\alpha_1)^{1/T}$. Here, α_0 is the initial injection strength, α_1 is the final injection strength, and t denotes the current decoding step and T is total number of steps we inject. This strongly promotes the target concept early in generation while preserving fluency and instruction following in later steps.

3.3 METRICS

AXBENCH Score. Following AXBENCH (Wu et al., 2025), we measure concept, fluency, and instruction-following scores. These validate whether steering causes the model to generate concept-related text (concept score) while also remaining relevant to the prompt (instruction score) and not producing degenerate text (fluency score). As an example, we describe how the concept score is evaluated: for each steered output, an LLM judge to assign a concept score in $\{0, 1, 2\}$. A score of 0 indicates that the target concept is absent; 1 indicates that the concept is present but expressed unnaturally or in an unintended manner; and 2 indicates that the concept is clearly and correctly expressed. The AXBENCH Score is then defined as the harmonic mean of the concept, fluency, and instruction-following scores.

Diversity Score. We introduce the diversity score to quantify how thoroughly a given representation covers the target concept. To compute it, we first extract the entities mentioned in each generated sentence (e.g., *cat*, *sparrow*) together with their associated category labels (e.g., *mammal*, *bird* for the ANIMALS concept). Second, we aggregate the extracted entities and categories across all samples and compute the normalized Shannon entropy at two levels: (i) *instance-level*, i.e. the diversity of distinct concept instances in any category (e.g., dog, lion, tiger), and (ii) *category-level*, which measures how broadly the generations span higher-level semantic categories within the concept (e.g., mammal, bird, reptile). The final Diversity Score (in $[0, 2]$) is the sum of the normalized entity- and category-level entropies (§B.2). Like recall, a high diversity score indicates broad concept coverage rather than concentration on a few instances or subcategories.[4]

[3] For 1D methods such as DiffMean-r1 and ReFT-r1, additive steering of the form $\boldsymbol{h}' = \boldsymbol{h} + \alpha \boldsymbol{s}$ is commonly used. In our experiments, we observed that additive steering substantially reduced output diversity, with generations frequently collapsing to a small set of entities from the target concept.

[4] Because instance entropy is normalized by the best possible diversity score, a method that causes the model to express the concept in only a few outputs will be penalized by both the diversity score *and* the concept score.

Table 2: Test scores for Gemma-3-27B. We report *Concept*, *Fluency*, *Instruction*, and *Diversity* scores, as well as the composite AXBENCH and MAXBENCH scores (metric definitions in §3.3). For each concept and method, we select the rank that achieves the highest mean MAXBENCH score on the validation set. We report mean ± standard deviation over the training splits.

Method	Concept	Fluency	Instruction	Diversity	AXBENCH	MAXBENCH
				= +	= HM(, ,)	= HM(, , ,)
Prompting	1.96 ± 0.04	1.99 ± 0.01	1.78 ± 0.06	1.54 ± 0.20	1.91 ± 0.02	1.79 ± 0.06
Diff Mean-r1	0.79 ± 0.32	1.29 ± 0.48	1.70 ± 0.07	1.42 ± 0.29	1.06 ± 0.27	1.10 ± 0.21
ReFT-r1	0.68 ± 0.24	1.37 ± 0.09	1.70 ± 0.06	**1.60** ± 0.27	1.04 ± 0.21	1.13 ± 0.21
MFA	0.76 ± 0.52	1.68 ± 0.26	1.76 ± 0.08	1.24 ± 0.32	1.06 ± 0.54	1.06 ± 0.49
SAE (top-k)	0.87 ± 0.34	1.46 ± 0.34	1.74 ± 0.07	1.38 ± 0.22	1.17 ± 0.23	1.21 ± 0.21
Factor Analysis	**1.33** ± 0.48	**1.73** ± 0.25	1.78 ± 0.04	1.44 ± 0.21	**1.53** ± 0.28	**1.49** ± 0.22
Schatten Probe	1.26 ± 0.50	1.68 ± 0.24	**1.79** ± 0.03	1.44 ± 0.13	1.49 ± 0.32	1.46 ± 0.25
Linear Probe	1.18 ± 0.53	1.72 ± 0.19	**1.79** ± 0.03	1.31 ± 0.16	1.44 ± 0.37	1.37 ± 0.27
PCA	1.31 ± 0.48	1.49 ± 0.44	1.78 ± 0.04	1.45 ± 0.23	1.39 ± 0.30	1.38 ± 0.22
DiffMean	1.29 ± 0.42	1.51 ± 0.50	**1.79** ± 0.03	1.45 ± 0.20	1.39 ± 0.28	1.39 ± 0.20

MAXBENCH Score. We now define the MAXBENCH score as the harmonic mean of the concept, fluency, instruction-following, *and diversity* scores. The MAXBENCH score rewards representations that both steer reliably toward the target concept and capture a broad range of values of that concept.

4 EXPERIMENTS

Models. We conduct experiments across three instruction-tuned model families, with varying sizes (Grattafiori et al., 2024; Team et al., 2025; Team, 2026). For each concept dataset, we extract hidden activations from later-to-middle layers, since semantic information is well formed at that stage of the model layers(§B.3) and it has been shown to work well in prior work (Turner et al., 2024; Rimsky et al., 2024). We use different layers at each model: 19 (Llama-3.1-8B); 17 (Gemma-3-1B); 53 (Gemma-3-27B); 27 (Qwen-3.5-4B). The choice of layer is constrained by the availability of trained SAEs, evaluating different layers is left to future work.

Datasets. We construct a dataset of six concepts: ANIMALS, COUNTRIES, VEHICLES, and PLANTS are hierarchical and unordered. DAYS is categorical and ordered, and YEARS is continuous and ordered. Hierarchical concepts are divided into subcategories. For example, ANIMALS comprises mammals, reptiles, *inter alia*, with each category containing multiple instances. Mammal, for instance, includes instances like *lion*, *beaver*, and *horse*. For each instance in each concept, we use GPT-5 (Singh et al., 2026) to generate 100 sentences ending with the target instance. We prompt models with these sentences before the instance token to form our datasets; see Fig. 1(a) and §C.1.

We validate that the models can perform the target-token prediction task by evaluating whether the intended instance is among the most likely next token given the prompt. Across all concepts, the correct entity appears among the top-5 tokens with 81% accuracy for the smallest model, while larger models achieve 97% accuracy on average. We provide behavioral results and examples in §C.2.

Dataset splits. We do not assume that a concept dataset contains all possible concept instances; for example, ANIMALS contains only a subset of all possible animals. To assess robustness to incomplete data, we evaluate each concept recovery method across multiple training splits. Each split contains differing subsets of the data. For hierarchical concepts, we use two types of splits: some splits hold out an entire subcategory, while others retain all subcategories but randomly exclude instances. We generate 10 splits in total, with 5 splits of each type. For non-hierarchical concepts such as DAYS and YEARS, we instead uniformly hold out a subset of instances.

Following Wu et al. (2025), we use prompts from AlpacaEval (Dubois et al., 2024) as inputs to the LM to perform interventions. For each prompt, we intervene on the model's hidden activations using a sampled point from the recovered concept geometry and observe the resulting model output. We randomly select 5 validation prompts for hyperparameter tuning and evaluate on a disjoint set of 5 test prompts. For each prompt, we sample 30 points from the recovered representation and perform 1 intervention with each sampled point; see §D.

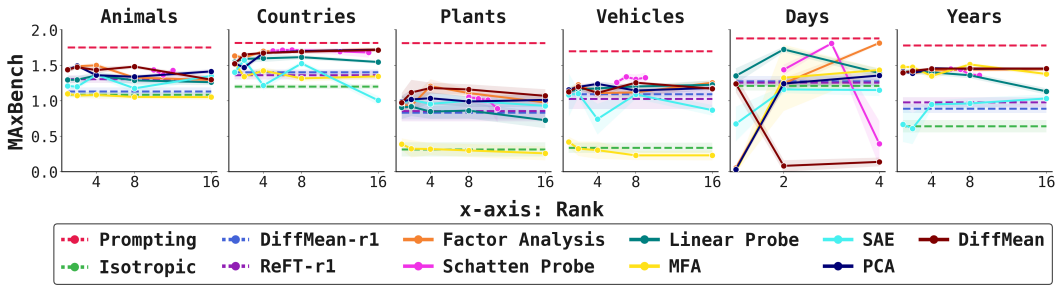


Figure 2: MAXBENCH score by rank for Gemma-3-27B. Confidence intervals are computed across training splits. Each point is the mean MAXBENCH score at a given (concept, method, rank); the shaded band is std. dev. Prompting, Isotropic, DiffMean-r1, and ReFT-r1 have no rank hyperparameter, so they are drawn as dashed lines. For Schatten Probes, we measure *effective* rank instead of manually setting its rank; we round to the nearest integer. Breakdown of individual metrics in §F.1.

Ranks. For methods that locate multi-dimensional subspaces, we sweep over ranks $k \in \{1, 2, 4, 8, 16\}$. The precise meaning of k depends on the method. For PCA, k is the number of leading singular directions retained in $\boldsymbol{L}$. For factor analysis, it is the number of latent factors; for MFA, k similarly denotes the number of latent factors per mixture component, and we train a separate model for each rank. For DiffMean and Linear Probes, k is the number of leading singular vectors retained in $\boldsymbol{L}$. For SAE, k is the top SAE features whose activations correlate most with the target concept. For Schatten Probe, we measure the effective rank of its learned weights (Roy & Vetterli, 2007). DiffMean-r1 and ReFT-r1 recover a single direction and therefore always have $k = 1$.

LLM Judge. We follow Wu et al. (2025) in using a large language model (LLM) judge to evaluate steered outputs. We use GPT-5.6-Luna (Singh et al., 2026) as our LLM judge. To assess the judge's alignment with human preferences, we report Krippendorff's α in §E.1. All prompts used for the LLM judges are in §E.2.

To evaluate diversity, we use an LLM to extract all concept instances mentioned across the steered outputs and (where applicable) to assign each instance to a subcategory. For hierarchical concepts (ANIMALS, COUNTRIES, PLANTS, VEHICLES), we provide the judge with the subcategories in our dataset (Tab. 6). Instances of the concept that do not belong to a predefined subcategory are assigned to an "other" subcategory. We then compute the normalized entropy of the resulting instance and category distributions, yielding the instance diversity and category diversity scores, respectively. For DAYS and YEARS, each instance is considered its own category.

5 RESULTS

Dimensionality. We first ask whether multinomial concepts are better represented by a single direction or by a higher-dimensional subspace. Tab. 2 summarizes the performance of the different concept localization methods. Multidimensional methods consistently achieve higher MAXBENCH scores than one-dimensional methods across concepts. This suggests at first that concepts with categorical non-binary structure may not be well captured by a single direction.

To evaluate whether this trend is attributable to the dimensionality of the concept representation, we observe performance across ranks for each method. Fig. 2 shows results for Gemma-3-27B across concepts. For affine methods, MAXBENCH scores are stable over the range of ranks we evaluate, with slight improvement as rank increases for some concepts. This suggests that much of the benefit of these methods can be obtained with quite low-dimensional subspaces, and that the *offset* $\boldsymbol{b}$ may explain much of the improvement. Results for other models are in §F.2.

Linear vs. Affine. To more directly measure how much the affine methods' performance depends on the offset $\boldsymbol{b}$ (rather than the bases $\boldsymbol{L}$), we compare affine and linear variants of each affine method, where linear variants have $\boldsymbol{b} = \boldsymbol{0}$. We observe (Fig. 3) that for Gemma-3-27B, affine variants achieve higher MAXBENCH scores than their linear counterparts. This also holds for other models (see §F.3). This extends prior findings that centering matters for steering (Jorgensen et al., 2023; Shafran et al., 2026) and concept erasure (Belrose et al., 2023).

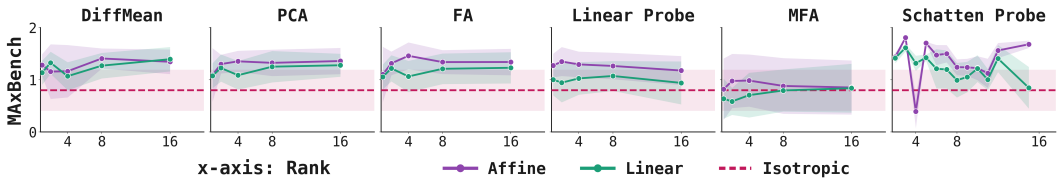


Figure 3: Comparison of multinomial concept localization methods for Gemma-3-27B with and without an offset. For the Schatten probe, we plot effective rank of the learned weight matrix as the p-norm parameter is varied from 0.1 to 0.9. For each method, we average the scores across all concepts and compute std. dev. across all dataset splits. Sampling around the concept mean offset is consistently better than assuming the subspace is centered at the origin. See §F.3 for other models.

Choice of basis. To assess $\boldsymbol{L}$'s contribution, we include a baseline that samples isotropically from the full representation space around an offset $\boldsymbol{b}$ defined by the mean of all concept-related activations. For Gemma-3-27B, isotropic sampling performs significantly worse than sampling within $\boldsymbol{L}$, suggesting that both the offset *and* basis contribute significantly to effective concept recovery. Interestingly, for smaller models, isotropic sampling sometimes outperforms linear methods, suggesting that the offset determines more of a concept representation's quality for these models.

Geometry type. Methods sharing a geometry type often perform similarly. For a given model, affine methods tend to have comparable MAXBENCH scores; directional methods have significantly lower MAXBENCH scores than affine subspaces, but perform comparably to each other. Thus, most of the performance is explained by the geometry type rather than properties of the exact methods.

Prompting. We include a prompting baseline where the model is instructed to incorporate the target concept when answering each AlpacaEval prompt, without intervening on the model's hidden activations (see §G for examples). For most models, prompting outperforms all representation-based methods (§F.4). An exception is Gemma-3-1B, where representation-based methods consistently outperform prompting. Thus, beyond a certain capability threshold, prompting remains a strong and simple baseline; this agrees with the findings of Wu et al. (2025) on binary concepts.

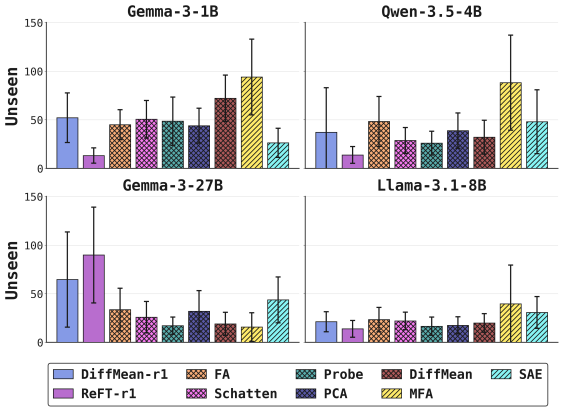


Figure 4: Unseen instances per model and method. We plot mean ± std across training splits at the best rank per method. Additional results in §F.5.

Non-Linear Methods. We additionally evaluate non-linear manifolds discovered via spline fitting (Wurgaft et al., 2026). Because this method requires ordered and ideally equidistant concept values, we apply it only to the DAYS concept. We evaluate spline fitting with both complete and incomplete concept data, where the latter is constructed by withholding a subset of days from the data used to fit the spline. Results for DAYS are summarized in Tab. 3. Spline fitting using the complete set of concept values outperforms spline fitting when some values are held out. More broadly, spline fitting with complete data achieves the highest overall performance among the methods evaluated on DAYS.

Table 3: MAXBENCH score on DAYS (averaged across models). Highest score per model is bolded.

	Gemma-3-1B	Gemma-3-27B	Llama-3.1-8B	Qwen-3.5-4B
Prompting	0.76 ± 0.06	1.88 ± 0.01	$\mathbf{1.86 \pm 0.01}$	$\mathbf{1.84 \pm 0.01}$
DiffMean-r1	1.04 ± 0.04	1.28 ± 0.07	1.27 ± 0.10	0.50 ± 0.10
ReFT-r1	0.93 ± 0.14	1.26 ± 0.04	0.98 ± 0.19	0.66 ± 0.10
MFA	1.16 ± 0.04	1.43 ± 0.02	0.02 ± 0.05	1.55 ± 0.03
SAE (top-k)	0.61 ± 0.35	1.16 ± 0.05	1.23 ± 0.05	0.46 ± 0.08
Factor Analysis	1.83 ± 0.01	1.82 ± 0.01	1.79 ± 0.01	1.72 ± 0.02
Schatten Probe	1.84 ± 0.02	1.81 ± 0.02	1.81 ± 0.01	1.74 ± 0.01
Linear Probe	1.78 ± 0.03	1.73 ± 0.04	1.77 ± 0.02	1.72 ± 0.02
PCA	1.82 ± 0.03	1.36 ± 0.06	1.77 ± 0.03	1.71 ± 0.02
DiffMean	$\mathbf{1.86 \pm 0.02}$	1.24 ± 0.12	1.80 ± 0.01	1.71 ± 0.02
Spline (missing days)	1.65 ± 0.02	1.79 ± 0.02	1.61 ± 0.07	1.66 ± 0.02
Spline	1.85 ± 0.12	$\mathbf{1.90 \pm 0.02}$	1.64 ± 0.05	1.72 ± 0.02

Generalization beyond concept data. How well do concept representations generalize beyond the instances used to derive them? To investigate, we provide the LLM judge with all concept instances and taxonomy labels observed in the training set (from the respective split) and ask it to classify each generated concept instance as either *seen*, if it appears in the training data, or *unseen* otherwise. We show the number of unseen instances in Fig. 4. All methods often yield unseen instances, indicating that the concept representations can generalize beyond the examples provided during training. Notably, Gaussian and unsupervised methods, particularly Factor Analysis, MFA, and SAEs, produce unseen instances relatively frequently. For Gemma-3-27B, rank-1 methods also generate a surprisingly large number of unseen instances.

6 Related Work

Concept geometry. Early concept geometry studies operated under the Linear Representation Hypothesis (Park et al., 2023), which assumes that concepts are linear and typically one-dimensional subspaces of LM activations. More recent work has proposed multidimensional concepts (Gurnee et al., 2026; Engels et al., 2025; Park et al., 2025; 2026; Vielhaben et al., 2023), which can be non-linear, and include geometries such as convex hulls (Fel et al., 2026b), polytopes (Park et al., 2025), vector magnitudes (rather than directions; Csordás et al., 2024), dense features (Sun et al., 2025; Lubana et al., 2026) or manifolds (Chung et al., 2018; Chung & Abbott, 2021; Modell et al., 2025; Kantamneni & Tegmark, 2025; Engels et al., 2025; Gurnee et al., 2026; Yocum et al., 2025; Karkada et al., 2026; Feucht et al., 2026; Wurgaft et al., 2026; Bhalla et al., 2026; Sarfati et al., 2026; Fel et al., 2026a). We focus extensively on affine subspaces, as they lie at an intermediate point of complexity: they enable one to go beyond single dimensions and beyond subspaces centered at the origin, but retain the simplicity and tractability of linear methods.

Evaluating concept representations. Given a concept representation, how can one verify its quality? Prior work has proposed correlational and causal metrics for evaluating their quality. Some evaluate geometric goodness-of-fit (Tiblias et al., 2026), whereas others evaluate counterfactual interventions on downstream behavior (Chen et al., 2025; Huang et al., 2024; Ding et al., 2026; Engels et al., 2025; Mueller et al., 2025; Shafran et al., 2026; Wu et al., 2025; Lee et al., 2026). Others still use toy settings with known ground truths (Gupta et al., 2024). These works evaluate whether a concept was leveraged after steering or whether a known pre-specified value was generated, but not whether the full concept space was covered; our diversity score fills this gap. These works also focus largely on binary concepts; our work specifically focuses on multinomial concepts.

Multidimensional/Multinomial concept steering. Most steering approaches assume that concepts correspond to single directions in activation space, and thus intervene by adding a scaled steering vector to the hidden state (Li et al., 2023; Zou et al., 2025; Turner et al., 2024; Subramani et al., 2022; Marks & Tegmark, 2024; You et al., 2026). Engels et al. (2025) instead steer by replacing the angular coordinate corresponding to one value (e.g., Monday) with another in a circular latent space, and then reconstruct the activation; this enables steering periodic variables like days of the week. Shafran et al. (2026) propose steering based on a Mixture of Factor Analyzers (MFA) decomposition of activation space, which models activations as belonging to local Gaussian regions with well-defined centroids and associated low-dimensional subspaces. Our work builds on this growing recognition that concepts in language models often exhibit intrinsically multi-dimensional structure. We propose a new framework to evaluate steering on multinomial concepts, only explored preliminarily in prior work (Wurgaft et al., 2026).

7 Conclusion

We have proposed a sampling-based framework for extrinsically evaluating multinomial concept representations. Based on this framework, we propose MaxBench to evaluate multinomial and multidimensional concept localization methods. Using MaxBench, we observe that affine methods consistently outperform rank-one or linear subspaces, and that manifold steering is a particularly strong method. Nonetheless, for larger models, no method consistently outperforms prompting. These findings highlight that the assumptions underlying concept localization methods may need to change as the field moves toward the study of increasingly complex concept representations, and that more appropriate geometric assumptions will accordingly grow increaisngly important.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets, implementing methods, cleaning and reformatting datasets, and to design or provide feedback on research methodology/experiments. We have not used generative AI tools to support qualitative and thematic data analysis nor for interpreting results. The following are not applicable for our work: help develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, assistance with translation. Additionally, we used generative AI tools to edit code, summarize existing literature, discover research topics or identify gaps, brainstorm, source/search for information, edit a research paper to improve readability, and identify relevant literature. We have reviewed all AI-assisted work and outputs generated by LLMs. LLM generated code was verified and tested for correctness. Literature survey, including summaries, suggested references, and any information retrieved with AI assistance, were also fully reviewed and verified. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We do not foresee any ethical issues directly resulting from this work. Interpretability research has uncovered many safety-relevant concept representations, including those related to refusal, truthfulness, and (mis)alignment. While this study does not directly engage with such concepts, this benchmark could indirectly support the development of methods better suited to locating these representations (for beneficial or harmful purposes).

REPRODUCIBILITY STATEMENT

We have provided details regarding our hyperparameter sweeps, datasets, experimental settings, LLM prompts, and other details that will support reproducibility in the Appendix. As this is an evaluation framework and benchmark that we intend to be extensible, we will publicly release all data in the benchmark and all code needed to replicate these experiments (and enable extensions).

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes, 2017. URL https://openreview.net/forum?id=ryF7rTqgl.

Dana Arad, Aaron Mueller, and Yonatan Belinkov. SAEs are good for steering – if you select the right features. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10241–10259, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.519. URL https://aclanthology.org/2025.emnlp-main.519/.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=awIpKpwTwF.

Usha Bhalla, Thomas Fel, Can Rager, Sheridan Feucht, Tal Haklay, Daniel Wurgaft, Siddharth Boppana, Matthew Kowal, Jack Merullo, Atticus Geiger, and Ekdeep Singh Lubana. Do sparse autoencoders capture concept manifolds? In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=euoAoOflaN.

Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf.

Dan Braun, Lucius Bushnaq, Stefan Heimersheim, Jake Mendel, and Lee Sharkey. Interpretability in parameter space: Minimizing mechanistic description length with attribution-based parameter decomposition, 2025. URL https://arxiv.org/abs/2501.14926.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Kai Chen, Zihao He, Taiwei Shi, and Kristina Lerman. STEER-BENCH: A benchmark for evaluating the steerability of large language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 18327–18355, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.925. URL https://aclanthology.org/2025.emnlp-main.925/.

SueYeon Chung and L.F. Abbott. Neural population geometry: An approach for understanding biological and artificial neural networks. *Current Opinion in Neurobiology*, 70:137–144, October 2021. ISSN 0959-4388. doi: 10.1016/j.conb.2021.10.010. URL http://dx.doi.org/10.1016/j.conb.2021.10.010.

SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Phys. Rev. X*, 8:031003, Jul 2018. doi: 10.1103/PhysRevX.8.031003. URL https://link.aps.org/doi/10.1103/PhysRevX.8.031003.

Róbert Csordás, Christopher Potts, Christopher D Manning, and Atticus Geiger. Recurrent neural networks learn to store and generate sequences using non-linear representations. In *The 7th BlackboxNLP Workshop*, 2024. URL https://openreview.net/forum?id=NUQeYgg8x4.

Zikang Ding, Qiying Hu, Yi Zhang, Hongji Li, Junchi Yao, Hongbo Liu, and Lijie Hu. Faithsteer-bench: A deployment-aligned stress-testing benchmark for inference-time steering, 2026. URL https://arxiv.org/abs/2603.18329.

Yann Dubois, Percy Liang, and Tatsunori Hashimoto. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=CybBmzWBX0.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Joshua Engels, Eric J Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=d63a4AM4hb.

Thomas Fel, Matthew Kowal, Mozes Jacobs, Dron Hazra, Usha Bhalla, Lee Sharkey, Lucius Bushnaq, Satchel Grant, Tal Haklay, Thomas Icard, Can Rager, Michael Pearce, Daniel Wurgaft, Aiden Swann, Fenil Doshi, Siddharth Boppana, Curt Tigges, Nick Cammarata, Thomas Serre, Vasudev Shyam, Owen Lewis, Thomas McGrath, Jack Merullo, Ekdeep Singh Lubana, and Atticus Geiger.

Structuring sparsity: Block-sparse featurizers capture visual concept manifolds, 2026a. URL https://arxiv.org/abs/2606.25234.

Thomas Fel, Binxu Wang, Michael A. Lepori, Matthew Kowal, Andrew Lee, Randall Balestriero, Sonia Joseph, Ekdeep Singh Lubana, Talia Konkle, Demba E. Ba, and Martin Wattenberg. Into the rabbit hull: From task-relevant concepts in DINO to minkowski geometry. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=pbbRmnFkjG.

Sheridan Feucht, Tal Haklay, Usha Bhalla, Daniel Wurgaft, Can Rager, Jack Merullo, Thomas McGrath, Raphaël Sarfati, Owen Lewis, Ekdeep Singh Lubana, Thomas Fel, and Atticus Geiger. Arithmetic in the wild: Llama uses standard addition to reason about cyclic concepts. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=92qkFSqoqk.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Paris Giampouras, Rene Vidal, Athanasios Rontogiannis, and Benjamin Haeffele. A novel variational form of the schatten-p quasi-norm. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21453–21463. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f53eb4122d5e2ce81a12093f8f9ce922-Paper.pdf.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami,

Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Rohan Gupta, Iván Arcuschin, Thomas Kwa, and Adrià Garriga-Alonso. Interpbench: Semi-synthetic transformers for evaluating mechanistic interpretability techniques. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=R9gR9MPuD5.

Wes Gurnee, Emmanuel Ameisen, Isaac Kauvar, Julius Tarng, Adam Pearce, Chris Olah, and Joshua Batson. When models manipulate manifolds: The geometry of a counting task, 2026. URL https://arxiv.org/abs/2601.04480.

Jing Huang, Zhengxuan Wu, Christopher Potts, Mor Geva, and Atticus Geiger. RAVEL: Evaluating interpretability methods on disentangling language model representations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8669–8687, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.470. URL https://aclanthology.org/2024.acl-long.470/.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Ole Jorgensen, Dylan Cope, Nandi Schoots, and Murray Shanahan. Improving activation steering in language models with mean-centring, 2023. URL https://arxiv.org/abs/2312.03813.

Shruti Joshi, Andrea Dittadi, Sébastien Lachapelle, and Dhanya Sridhar. Sparse shift autoencoders for identifying concepts from large language model activations, 2026. URL https://arxiv.org/abs/2502.12179.

Subhash Kantamneni and Max Tegmark. Language models use trigonometry to do addition. In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*, 2025. URL https://openreview.net/forum?id=CqViN4dQJk.

Dhruva Karkada, Daniel James Korchinski, Andres Nava, Matthieu Wyart, and Yasaman Bahri. Symmetries in language statistics shape the geometry of model representations. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=XQeMPEkfdd.

Andrew Lee, Yonatan Belinkov, Fernanda Viégas, and Martin Wattenberg. Decomposing query-key feature interactions using contrastive covariances. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=t5cfrJfuzD.

John M. Lee. *Introduction to Smooth Manifolds*, volume 218 of *Graduate Texts in Mathematics*. Springer, New York, 2 edition, 2012. ISBN 978-1-4419-9982-5. doi: 10.1007/978-1-4419-9982-5.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=aLLuYpn83y.

Ekdeep Singh Lubana, Can Rager, Sai Sumedh R. Hindupur, Valérie Costa, Oam Patel, Sonia Krishna Murthy, Thomas Fel, Greta Tuckute, Daniel Wurgaft, Eric Bigelow, Demba E. Ba, Melanie Weber, and Aaron Mueller. Priors in time: Missing inductive biases for language model interpretability. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=4J2e3nWiC8.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=aajyHYjjsk.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL https://arxiv.org/abs/1301.3781.

George A. Miller. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992. URL https://aclanthology.org/H92-1116/.

Alexander Modell, Patrick Rubin-Delanchy, and Nick Whiteley. The origins of representation manifolds in large language models, 2025. URL https://arxiv.org/abs/2505.18235.

Aaron Mueller, Atticus Geiger, Sarah Wiegreffe, Dana Arad, Iván Arcuschin, Adam Belfki, Yik Siu Chan, Jaden Fried Fiotto-Kaufman, Tal Haklay, Michael Hanna, Jing Huang, Rohan Gupta, Yaniv Nikankin, Hadas Orgad, Nikhil Prakash, Anja Reusch, Aruna Sankaranarayanan, Shun Shao, Alessandro Stolfo, Martin Tutek, Amir Zur, David Bau, and Yonatan Belinkov. MIB: A mechanistic interpretability benchmark. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=sSrOwve6vb.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. In Yonatan Belinkov, Sophie Hao, Jaap Jumelet, Najoung Kim, Arya McCarthy, and Hosein Mohebbi (eds.), *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 16–30, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.blackboxnlp-1.2. URL https://aclanthology.org/2023.blackboxnlp-1.2/.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Causal Representation Learning Workshop at NeurIPS 2023*, 2023. URL https://openreview.net/forum?id=T0PoOJg8cK.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=bVTM2QKYuA.

Kiho Park, Todd Nief, Yo Joong Choe, and Victor Veitch. The information geometry of softmax: Probing and steering. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=BrIGLICBWy.

Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, pp. 606–610, 2007.

Raphaël Sarfati, Eric Bigelow, Daniel Wurgaft, Jack Merullo, Atticus Geiger, Owen Lewis, Tom McGrath, and Ekdeep Singh Lubana. The shape of beliefs: Geometry, dynamics, and interventions along representation manifolds of language models' posteriors, 2026. URL https://arxiv.org/abs/2602.02315.

Or David Shafran, Shaked Ronen, Omri Fahn, Shauli Ravfogel, Atticus Geiger, and Mor Geva. From directions to regions: Decomposing activations in language models via local geometry. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Tz7n7pX9cO.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone,

Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, CJ Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, DJ Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang, Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang, Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Y. Guan, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, RJ Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song Mei, Sonya Egoian, Spencer Kim, Spug Golden, SQ Mah, Steph Lin, Stephen Imm, Steve Sharpe, Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer,

Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomek Korbak, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

C. Spearman. "general intelligence" objectively determined and measured. Studies in individual differences: The search for intelligence., 1961.

Nishant Subramani, Nivedita Suresh, and Matthew Peters. Extracting latent steering vectors from pretrained language models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 566–581, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.48. URL https://aclanthology.org/2022.findings-acl.48/.

Xiaoqing Sun, Alessandro Stolfo, Joshua Engels, Ben Peng Wu, Senthooran Rajamanoharan, Mrinmaya Sachan, and Max Tegmark. Dense SAE latents are features, not bugs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=p8lKcNkJRi.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste

Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Qwen Team. Qwen3.5-omni technical report, 2026. URL https://arxiv.org/abs/2604.15804.

Federico Tiblias, Irina Bigoulaeva, Jingcheng Niu, Simone Balloccu, and Iryna Gurevych. Hypothesis-driven feature manifold analysis in LLMs via supervised multi-dimensional scaling. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=vCKZ40YYPr.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Johanna Vielhaben, Stefan Bluecher, and Nils Strodthoff. Multi-dimensional concept discovery (MCD): A unifying framework with completeness guarantees. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=KxBQPz7HKh.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=fykjplMc0V.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Axbench: Steering LLMs? even simple baselines outperform sparse autoencoders. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=K2CckZjNy0.

Daniel Wurgaft, Can Rager, Matthew Kowal, Sheridan Feucht, Usha Bhalla, Tal Haklay, Eric Bigelow, Raphaël Sarfati, Jack Merullo, Noah Goodman, Thomas Fel, Atticus Geiger, and Ekdeep Singh Lubana. Manifold steering reveals the shared geometry of neural network representation and behavior. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=pf3HKAGT2M.

Julian Yocum, Cameron Allen, Bruno Olshausen, and Stuart Russell. Neural manifold geometry encodes feature fields. In *NeurIPS 2025 Workshop on Symmetry and Geometry in Neural Representations*, 2025. URL https://openreview.net/forum?id=MwU86qfCTW.

Zejia You, Chunyuan Deng, and Hanjie Chen. Spherical steering: Geometry-aware activation rotation for language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=9PPeJsVLdB.

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 Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2025. URL https://arxiv.org/abs/2310.01405.

APPENDIX CONTENTS

A Concept Localization Methods **20**

A.1 Detailed Descriptions of Methods 20

A.2 SAE Feature Selection 21

A.3 MFA Gaussian Selection 23

B MAXBENCH Framework **23**

B.1 Hyperparameter Tuning for Steering 23

B.2 Diversity Scores and Normalization 24

B.3 Layerwise Analysis 25

C Datasets **26**

C.1 Concepts 26

C.2 Behavioral Performance 27

D Examples for Sampling-Based Steering Output **28**

E LLM-Judge **31**

E.1 Evaluating Agreement between LLM Judge and Human 31

E.2 LLM Judge Prompts 31

F Supplementary Experiments **40**

F.1 Metric Breakdown for MAxBench Score 40

F.2 Metric Meta-evaluation 43

F.3 Affine vs Linear Comparison 43

F.4 Sampling Evaluation Results across Models 44

F.5 Unseen Instances Analysis 47

G Prompting Baseline **48**

A CONCEPT LOCALIZATION METHODS

A.1 DETAILED DESCRIPTIONS OF METHODS

DiffMean-r1 (Marks & Tegmark, 2024). DiffMean-r1 (commonly known as *difference in means* in literature, but we call it DiffMean-r1 to differentiate it from our *multidimensional* difference in means method) learns a concept representation from binary concept labels by computing the difference between the mean activations of the positive and negative examples. Let $\boldsymbol{X}^+$ and $\boldsymbol{X}^-$ denote the activation matrices corresponding to the positive and negative concept examples, respectively. The localized concept direction is given by $\mathbb{G}_{\text{DiffMean}} = \boldsymbol{\mu}^+ - \boldsymbol{\mu}^-$, where $\boldsymbol{\mu}^+ = \frac{1}{|\boldsymbol{X}^+|}\sum_{\boldsymbol{x}\in\boldsymbol{X}^+}\boldsymbol{x}$ and $\boldsymbol{\mu}^- = \frac{1}{|\boldsymbol{X}^-|}\sum_{\boldsymbol{x}\in\boldsymbol{X}^-}\boldsymbol{x}$. The localized representation is therefore a one-dimensional affine subspace of the form $\mathbb{G}_{\text{DiffMean}} = \boldsymbol{L}q + \boldsymbol{b} \mid q \in \mathbb{R}$, where $\boldsymbol{b} = \boldsymbol{0}$, and $\boldsymbol{L}$ is normalized to unit norm.

ReFT-r1 (Wu et al., 2025; 2024): ReFT-r1 learns concept representation from supervised concept labels by jointly optimizing concept detection and representation-level steering. During training, it learns a single 1D subspace $\mathbb{G}_{\text{ReFT}} \in \mathbb{R}^d$ that is used both to detect the presence of the concept and to intervene on the hidden representations. Complete training objective is present in (Wu et al., 2025; 2024). Similar to DiffMean-r1 localized representation is therefore a one-dimensional direction of the form $\mathbb{G}_{\text{ReFT}} = \boldsymbol{L}q + \boldsymbol{b} \mid q \in \mathbb{R}$, where $\boldsymbol{b} = \boldsymbol{0}$ and $\boldsymbol{L} = \mathbb{G}_{\text{ReFT}}$ is normalized to unit norm. Steering for ReFT-r1 is also similar to DiffMean-r1.

Note that the concept dataset used for REFT-r1 differs slightly from those used by the other methods. REFT-r1 requires paired training examples of two types: responses that answer an instruction while incorporating the target concept, and responses to the instruction that do not incorporate the concept. The representation is then learned from this dataset. We generate this dataset using GPT-5 Singh et al. (2026). For example, for the concept ANIMALS, given the instruction *'Write one sentence about a busy city"*, a positive response could be *'The busy city streets were filled with people, cars, and pigeons circling above the buildings,"* while a negative response could be *"The busy city streets were filled with people and cars moving between tall buildings."*

Linear Probe (Alain & Bengio, 2017): Linear Probe learns a linear classifier to predict concept labels from hidden activations. Given activation vectors $\boldsymbol{x}^{(i)}$ and corresponding concept labels $y^{(i)} \in 1, \ldots, |C|$, it learns classifier parameters $\boldsymbol{W} \in \mathbb{R}^{d\times|C|}$ and $\boldsymbol{b} \in \mathbb{R}^{|C|}$ by minimizing the cross-entropy loss. We then perform the singular value decomposition $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ and localize the concept basis as the top k left singular vectors, $\boldsymbol{L} = [\boldsymbol{u}_1|\cdots|\boldsymbol{u}_k] \in \mathbb{R}^{d\times k}$. The learned classifier bias $\boldsymbol{b}$ cannot directly serve as the affine offset, since it lies in the $|C|$-dimensional classifier output space rather than the d-dimensional residual-stream space. We therefore set the affine offset to the global activation mean, $\boldsymbol{\mu}_C = \frac{1}{m}\sum_{i=1}^m \boldsymbol{x}^{(i)}$. The localized representation is therefore a k-dimensional affine subspace of the form $\mathbb{G}_{\text{Probe}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{b} \mid \boldsymbol{q} \in \mathbb{R}^k$, where $\boldsymbol{b} = \boldsymbol{\mu}_C$.

We treat the localized representation as an affine subspace and sample intervention points using the Gaussian sampling procedure described in §3.2.

PCA (Pearson, 1901): Given activation vectors $\{\boldsymbol{x}^{(i)}\}_{i=1}^m$, we first compute the mean activation $\boldsymbol{\mu}_C = \frac{1}{m}\sum_{i=1}^m \boldsymbol{x}^{(i)}$ and center the activations. We then perform singular value decomposition of the centered activation matrix (stack of $\{\boldsymbol{x}^{(i)}\}_{i=1}^m$), $\tilde{\boldsymbol{X}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, and retain the first k right singular vectors to form the basis $\boldsymbol{L} = [\boldsymbol{v}_1|\cdots|\boldsymbol{v}_k] \in \mathbb{R}^{d\times k}$. The resulting concept representation is a k-dimensional affine subspace centered at the mean activation, $\mathbb{G}_{\text{PCA}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{\mu}_C \mid \boldsymbol{q} \in \mathbb{R}^k$.

Factor Analysis (Spearman, 1961): Factor Analysis models the observed activations as arising from a low-dimensional latent representation corrupted by gaussian noise. Specifically, activations are modeled as $\boldsymbol{x}^{(i)} = \boldsymbol{\mu} + \boldsymbol{L}\boldsymbol{z}^{(i)} + \epsilon^{(i)}$, where $\boldsymbol{\mu} \in \mathbb{R}^d$ is the mean activation, $\boldsymbol{L} \in \mathbb{R}^{d\times k}$ is the factor loading matrix, $\boldsymbol{z}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_k)$ are latent factors, and $\epsilon^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Psi})$ is Gaussian noise with diagonal covariance $\boldsymbol{\Psi} = \text{diag}(\psi_1, \ldots, \psi_d)$. The parameters $\boldsymbol{L}, \boldsymbol{\Psi}, \boldsymbol{\mu}$ are estimated using the Expectation-Maximization (EM) algorithm. The localized representation is a k-dimensional affine subspace of the form $\mathbb{G}_{\text{FA}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{\mu} \mid \boldsymbol{q} \in \mathbb{R}^k$. We sample intervention points from the localized affine subspace using the gaussian sampling procedure as we do for other methods.

DiffMean DiffMean-Pairs extends DiffMean-r1 (Marks & Tegmark, 2024) to multi-class concepts by computing pairwise difference vectors between class centroids. Let $C = c_1, \ldots, c_{|C|}$ denote the set of concept classes and let $\boldsymbol{\mu}_{c_j}$ be the mean activation of class c_j. For every pair of classes (c_i, c_j) with $i < j$, we compute the difference vector $\boldsymbol{d}_{i,j} = \boldsymbol{\mu}_{c_i} - \boldsymbol{\mu}_{c_j}$. Stacking all pairwise difference vectors forms the matrix $\boldsymbol{D} \in \mathbb{R}^{d \times p}$, where $p = \binom{|C|}{2}$. We then perform singular value decomposition, $\boldsymbol{D} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, and retain the first k left singular vectors to form the basis $\boldsymbol{L} = [\boldsymbol{u}_1 | \cdots | \boldsymbol{u}_k] \in \mathbb{R}^{d \times k}$. The affine offset is set to the global activation mean, $\boldsymbol{\mu}_C = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}^{(i)}$.

The localized representation is therefore a k-dimensional affine subspace of the form $\mathbb{G}_{\text{DMP}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{\mu}_C \mid \boldsymbol{q} \in \mathbb{R}^k$. We treat the localized representation as an affine subspace and sample intervention points using the Gaussian sampling procedure described above.

Schatten Probe. Schatten Probe extends the linear probe by encouraging the classifier weight matrix to be low-rank through approximate Schatten-p norm regularization. The classifier weight matrix is factorized as $\boldsymbol{W} = \boldsymbol{A}\boldsymbol{B}^\top$, where $\boldsymbol{A} \in \mathbb{R}^{d \times k}$ and $\boldsymbol{B} \in \mathbb{R}^{|C| \times k}$. The parameters are learned by minimizing the cross-entropy loss together with a Schatten p-norm regularizer inspired by (Giampouras et al., 2020; Braun et al., 2025) that encourages the probe's weights to be low-rank. After training, the weight matrix is reconstructed as $\boldsymbol{W} = \boldsymbol{A}\boldsymbol{B}^\top$ and decomposed via singular value decomposition, $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$. We retain the first k_{eff} left singular vectors, where k_{eff} is determined by the effective rank (Roy & Vetterli, 2007) of $\boldsymbol{W}$, and set the affine offset to the global activation mean $\boldsymbol{\mu}_C = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}^{(i)}$.

The localized representation is therefore a k_{eff}-dimensional affine subspace of the form $\mathbb{G}_{\text{Schatten}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{\mu}_C \mid \boldsymbol{q} \in \mathbb{R}^{k_{\text{eff}}}$, where $\boldsymbol{L} = [\boldsymbol{u}_1 | \cdots | \boldsymbol{u}_{k_{\text{eff}}}]$.

MFA (Shafran et al., 2026). Mixture of Factor Analyzers (MFA) is an unsupervised method that models the activation space as a mixture of low-dimensional gaussian subspaces. Following (Shafran et al., 2026), we train MFA on activations from the PILE (Gao et al., 2020) dataset. We train one MFA model with 9,000 Gaussian components and defer the other details of model fitting to the original work. Unlike conept dataset, we take activation at every token of PILE while training MFAs.

To identify the component corresponding to a target concept, we use the concept dataset after training. For each activation $\boldsymbol{x}$, MFA computes the posterior responsibility $r_k(\boldsymbol{x}) = p(k \mid \boldsymbol{x})$ for every Gaussian component k. More details about how to compute responsibilities are in §A.3. Given positive concept activations $\mathcal{P}$ and negative activations $\mathcal{N}$, we compute a concept score for each Gaussian as $s_k = \frac{1}{|\mathcal{P}|}\sum_{\boldsymbol{x} \in \mathcal{P}} r_k(\boldsymbol{x}) - \frac{1}{|\mathcal{N}|}\sum_{\boldsymbol{x} \in \mathcal{N}} r_k(\boldsymbol{x})$. Intuitively, a high score indicates that the component is consistently activated by the target concept while assigning low responsibility to unrelated activations. We rank all Gaussian components according to s_k and select the highest-scoring component.

The localized representation is therefore one single gaussian component and we treat this as k-dimensional affine subspace $\mathbb{G}_{\text{MFA}} = \boldsymbol{L}\boldsymbol{q} + \boldsymbol{\mu} \mid \boldsymbol{q} \in \mathbb{R}^k$, where $\boldsymbol{L}$ and $\boldsymbol{\mu}$ denote the factor loading matrix and centroid of the selected gaussian component.

SAE (Bricken et al., 2023). SAE represents concepts using sparse one-dimensional features learned by a pretrained sparse autoencoder. We encode activations using a pretrained SAE and identify the subset of features that best predicts a target concept. For each feature, we compute its Matthews Correlation Coefficient (MCC) with the binary concept labels and rank features according to this score. We then select the top k features together with their decoder vectors. We try two different strategies for sampling with SAE top-k features and select the best performing strategy. More details are in §A.2.

A.2 SAE Feature Selection

We do not train our own sparse autoencoders; for every model in our evaluation we use a publicly released SAE. For Llama-3.1-8B-Instruct, we use the Goodfire SAE trained at layer 19. For Gemma-3-1B and Gemma-3-27B, we use residual-stream SAEs from Gemma Scope 2 at layer 17 and Gemma Scope 2 at layer 53, respectively, using the 65k-width variants. For Qwen-3.5-4B, we use the Decode Research SAE trained at layer 27.

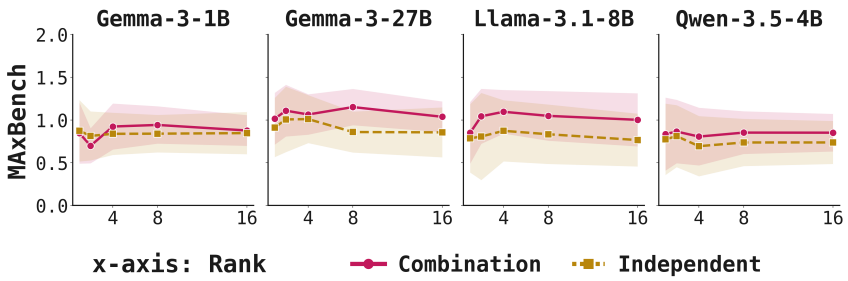


Figure 5: **Comparison of SAE sampling strategies across models.** Each subplot corresponds to one model. The solid red line shows the *combination* method, where the top-k SAE features are treated jointly as a basis for a subspace, while the dotted brown line shows *independent sampling*, where each sample is drawn along a single randomly selected feature direction. This is averaged over all 10 dataset splits and all concepts. Across models and ranks, combination sampling performs slightly better overall than independent sampling.

SAEs Bricken et al. (2023) are typically trained to produce one-dimensional feature directions, but individual features may or may not cleanly correspond to a target concept. Thus, it is important to select the right SAE features (Arad et al., 2025). Doing so requires addressing three questions: How should features be selected for a given concept dataset? How many features do we select? How do we sample from multiple features?

For feature selection, we use an MCC-based method (Joshi et al., 2026). We treat number of top features that correspond to a concept top-k as a hyperparameter and evaluate on a range of values. We then evaluate several strategies for sampling from the selected features. In the main Tab. 2, we report the best-performing feature-combination method. For a given sentence, we extract the residual-stream last token activation at the model's fixed layer. This activation is passed through the SAE encoder to obtain a sparse feature vector.

Feature selection. Given a set of positive (concept) and negative activations, we score every SAE feature by how well its activation separates the two classes. For each feature f we use different quantile thresholds and we binarize that feature's activations at the threshold and compute the Matthews Correlation Coefficient (MCC) against the ground-truth positive/negative labels:

$$\mathrm{MCC} = \frac{tp \cdot tn - fp \cdot fn}{\sqrt{(tp+fp)(tp+fn)(tn+fp)(tn+fn)}}.$$

Each feature's score is its *best* MCC over the 9 different thresholds. We then take the top k=16 features by this score. For each selected feature f, we take its SAE *decoder* column $W_{\mathrm{dec}}[f] \in \mathbb{R}^D$ as its direction in activation space. Decoder columns are unit-norm by construction so we do not orthogonalize the selected directions. We sweep rank $k \in \{1, 2, 4, 8, 16\}$.

Sampling from multiple features. We explore two ways of sampling from the top-k SAE features. First, we can treat the selected features independently: for each sample s_i, we randomly choose one of the top-k features and draw from a gaussian along that feature direction (the independent sampling method). Thus, each sample comes from a different SAE feature. The intuition is that if all selected features correspond to the same concept, then samples drawn from any of these directions should still remain within the concept. Second, we can treat the top-k feature directions jointly as basis vectors and sample from the subspace that they span (the combination method).

Independent Sampling. In this method, for each of the N_{samples} draws independently, we uniformly pick one of the top-k feature directions f_i , each carrying its own individually-calibrated standard deviation σ_{f_i} (more about how we do this for each type of geometry in §B.1) and sample a single point from that one feature's rank-1 subspace alone.

Combination. In this method we stack the top-k features to form a subspace by treating them individually as basis vectors. $B_k = [\, W_{\mathrm{dec}}[f_1] \mid \cdots \mid W_{\mathrm{dec}}[f_k] \,]$, : rank 1 is feature f_1 alone, rank 2 adds f_2, and so on. Results for comparing both these methods are in Fig. 5.

Empirically we observe that across models and ranks, combination sampling performs slightly better overall than independent sampling.

A.3 MFA Gaussian Selection

We train MFAs from scratch on the Pile (Gao et al., 2020), following the original implementation of Shafran et al. (2026). We use the same hyperparameters as the original implementation, except for the Gaussian subspace rank, which we vary from 1 to 16. To select top MFA components we do the following: Given positive (concept) and negative activations, we compute the soft responsibility of each component using the standard factor-analysis posterior, with temperature scaling parameter τ:

$$\mathrm{resp}_k(x) = \frac{\exp\big((\ell_k(x) + \log \pi_k)/\tau\big)}{\sum_{k'} \exp\big((\ell_{k'}(x) + \log \pi_{k'})/\tau\big)},$$

where $\ell_k(x)$ is the log-likelihood of x under Gaussian component k, and π_k is its mixture weight.

We then score each component according to how much more responsibility it assigns to positive examples than to negative examples:

$$\mathrm{score}_k = \overline{\mathrm{resp}_k}(x^+) - \overline{\mathrm{resp}_k}(x^-),$$

We rank all components by this score and retain the highest-scoring component, i.e., the component that is most selective for the target concept. Once a component (μ_k, W_k) is selected, we sample from it using the same convention as the other subspace methods in our benchmark.

B MaxBench Framework

B.1 Hyperparameter Tuning for Steering

Tuning hyperparams for (Schatten) probes. For each dataset split, we reserve a fixed validation set that is disjoint from the training set and use it for hyperparameter selection for both linear and schatten probes.

For schatten probes, we treat the Schatten-p norm parameter p as a hyperparameter and sweep ($p \in \{0.1, 0.2, \ldots, 0.9\}$). This produces multiple candidate schatten probes at each integer rank. For each rank, we retain the candidate that achieves the highest MaxBenchscore on the validation set. We sometimes refer to dataset split also as bootstrap.

Selecting σ scale Each concept localization method produces a basis (and, for the affine variant, an offset) to sample from. We always sample centered at the offset for affine subspaces; for linear subspaces, we center at origin. With this set, we still need to select the variance σ within which to sample for each subspace. Gaussian sampling with standard deviation of (for example) $\sigma = 0.5$ is meaningless in isolation, as it means something different in every model's activation space. We therefore calibrate σ per each model and also each subspace (individually for each rank).

For each model, we approximate the natural range of the activation norm. To do this, we take all 805 instructions from AlpacaEval that we use and extract the residual-stream activation at the last token giving a set of activation vectors $X = \{x_i\}_{i=1}^N \subset \mathbb{R}^D$, $N \approx 805$.

From X, we record the empirical distribution of activation norms $\|x_i\|_2$; in particular, we select minimum and maximum norm observed for model m. We then use thus min ($r_{\min}$) and max ($r_{\max}$) norms to select appropriate σ.

Standard-deviation selection. The appropriate sampling σ depends on the dimensionality of the localized subspace. A value of σ that produces reasonable samples at low rank can generate points with unrealistically large norms at higher rank. We therefore calibrate σ separately for each method and rank rather than using a single fixed value. For each concept and model, we consider a grid of $n = 10$ candidate standard deviations between $\sigma_{\min} = r_{\min}$ and the maximum activation norm observed for that model, $\sigma_{\max} = r_{\max}$.

For each model, we record the minimum and maximum activation norms observed in real model activations. We then create 10 candidate values of σ, spaced logarithmically between $r_{\min}$ and $r_{\max}$.

For every method and rank, and for both the affine and linear variants, we sample 500 points using each candidate value of σ. For each value, we compute the fraction of sampled points whose norm falls within the range of norms observed in real model activations, $[r_{\min}, r_{\max}]$.

We select the value of σ that produces the largest fraction of samples inside this range. If several values perform equally well, we choose the largest σ. This gives us a separate sampling scale for each method and rank, while ensuring that sampled activations remain within a realistic norm range.

We steer the model by interpolating between the original activation h and a sampled point s from the localized concept representation:

$$\boldsymbol{h}' = (1 - \alpha)\boldsymbol{h} + \alpha \boldsymbol{s}.$$

Here, α controls the steering magnitude. We perform a grid search over α and report results using the value that gives the best steering performance.

B.2 Diversity Scores and Normalization

To quantify the diversity of the steered outputs, we distinguish between diversity at the level of predefined categories and diversity at the level of individual instances. We measure both using normalized Shannon entropy.

For each prompt, we sample 30 steered outputs. Across the 5 prompts, this results in $N = 5 \times 30 = 150$ outputs in total. For a single steered output we detect every instance of the concept that it incorporates. We collect them for all N outputs in the multiset $\mathcal{I}$. Note that a single steered output sentence can contain zero, one or several instances.

Categorical diversity. Each observed instance $x \in \mathcal{I}$ is assigned to one of a predefined set of categories, plus a category which is added to collect instances that do not fall within those (e.g. dragons are fantasy animals, which is not a category in our definition of ANIMALS):

$$\mathrm{SubC} = \{c_1, \ldots, c_K, c_{\text{other}}\}, \qquad |\mathrm{SubC}| = K + 1.$$

The empirical probability of subcategory $c \in \mathrm{SubC}$ is

$$p_{\text{cat}}(c) = \frac{|\{x \in \mathcal{I} : x \text{ belongs to } c\}|}{|\mathcal{I}|}.$$

We define the categorical Shannon entropy as

$$H_{\text{cat}} = -\sum_{c \in \mathrm{SubC}} p_{\text{cat}}(c) \log p_{\text{cat}}(c).$$

The maximum possible categorical entropy is $\log K + 1$, attained when all subcategories occur with equal probability. We therefore define normalized categorical diversity as

$$D_{\text{cat}} = \frac{H_{\text{cat}}}{\log |\mathrm{SubC}|}.$$

Thus, $D_{\text{cat}} \in [0, 1]$, where $D_{\text{cat}} = 0$ corresponds to all observations belonging to a single subcategory, while $D_{\text{cat}} = 1$ corresponds to a uniform distribution over all predefined subcategories, including the "other" category.

Instance-level diversity. Categorical diversity does not capture variation between individual instances within the same subcategory. We therefore additionally measure diversity over the individual instances themselves. However, unlike in the category diversity setting, we do not necessarily know the complete support $|\mathcal{U}|$of a concepts unique instances. Indeed, we observe some new instances e.g. in Table 15, whose number can vary greatly. Therefore, we distinguish two ways of determining the size of the concept's support depending on that dataset, which differ only in the normalization constant.

Let $\mathcal{U} = \{x : x \in \mathcal{I}\}$ denote the set of unique instances observed in $\mathcal{I}$. For each unique instance $u \in \mathcal{U}$, its empirical probability is

$$p_{\text{inst}}(u) = \frac{|\{x \in \mathcal{I} : x = u\}|}{|\mathcal{I}|}.$$

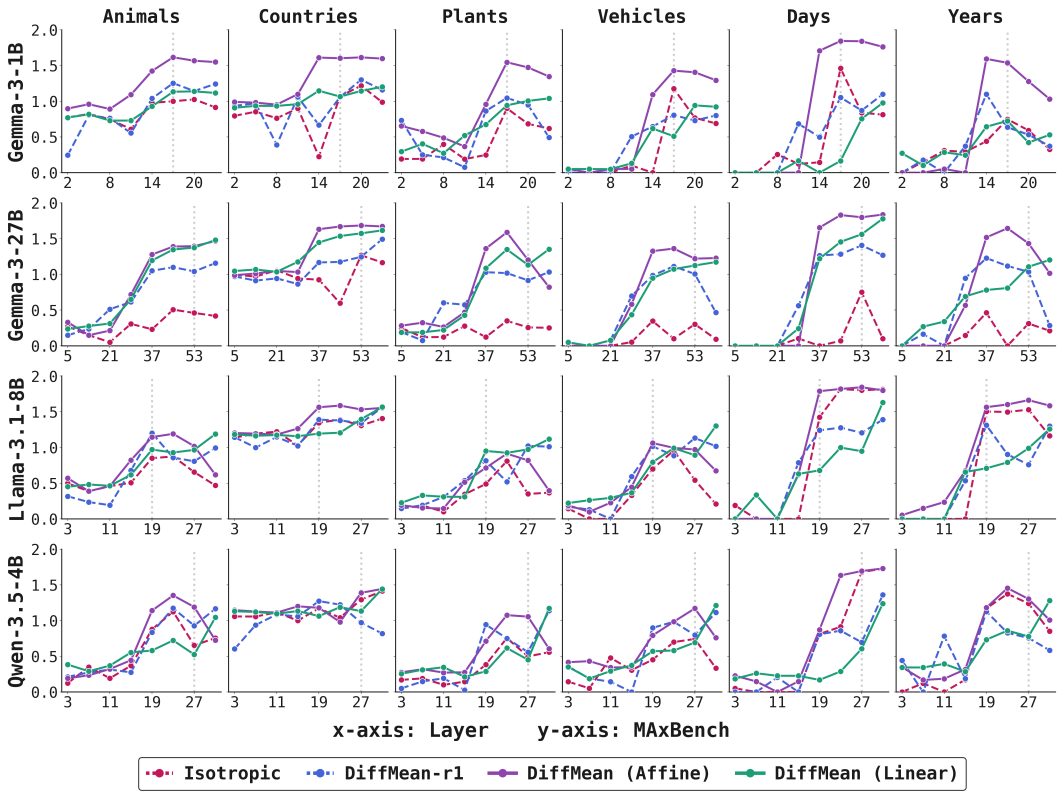


Figure 6: Layer sweeps across different concepts and all models considered in our work. For each model, we evaluate eight layers selected at approximately equal intervals throughout the network. At each layer, we evaluate four steering methods across all concepts. The vertical line indicates the layer we chose for our main analyses (largely based on the availability of SAEs or MFAs); while not always optimal, it is generally the case that the middle layers and upper-middle layers have largely comparable performance for many models. We generally operate at the upper-middle layers, and so we expect that trends we observe will be roughly stable within these layers.

The instance-level Shannon entropy is

$$H_{\text{inst}} = -\sum_{u \in \mathcal{U}} p_{\text{inst}}(u) \log p_{\text{inst}}(u).$$

For normalization we use

$$D_{\text{inst}} = \frac{H_{\text{inst}}}{\log Z}.$$

Where Z depends on the setting:

1. If the complete set $\mathcal{U}_{\max}$ containing possible instances of a concept **is known** and $|\mathcal{U}_{\max}| < N$, we use $Z = |\mathcal{U}_{\max}|$.
2. If the complete set $\mathcal{U}_{\max}$ containing possible instances of a concept **is unknown** , we estimate $Z = N$. Even though many instances can occur in a single output, so practically $|\mathcal{U}| > N$, we chose to set the normalization constant to $\log N$ for the datasets we consider. This choice is justified as on average we have fewer than one instance per output mentioned. If this does not hold true in practice, one might need to set the normalization constant larger than N, e.g. use $|\mathcal{U}|$.

B.3 Layerwise Analysis

We perform layer sweeps to identify appropriate intervention layers and to examine how the performance of different steering methods changes across layers. For this analysis, we consider representa-

tive methods from each method class: DiffMean-r1 as the one-dimensional method, DiffMean(pairs) affine as the affine method, DiffMean(pairs) linear as the linear method, and isotropic baseline. Additionally we run this experiment only using one data split. We use Qwen-3.6-27B Instruct tuned model as a judge for layer sweeps since using GPT-Luna-5.6 is not cost effective given the scale of this experiment. We report Krippendorff's α between gpt-5.6-luna and Qwen-3.6-27B in Tab. 4.

Across layers, that MAXBENCH score increases until middle layers and reamins mostly contant after that or reduces. Scores for the affine version of DiffMean is highest for most layers. We also find that isotropic steering performs well across a range of layers. Overall, the layers selected for our main experiments fall within regions where the corresponding methods achieve good steering performance. Results are in Fig. 6.

Table 4: Krippendorff's α between gpt-5.6-luna and Qwen-3.6-27B

Concept	**Fluency**	**Instruction**	n
$+0.7074$	$+0.6097$	$+0.9088$	280

C DATASETS

C.1 CONCEPTS

We generate 6 concept datasets and 1 control dataset. Some example prompts from our dataset are presented in Tab. 5 and different categories and entities per concept are in Table 6. Categories for ANIMALS are derived from WordNet (Miller, 1992), plants are classified according to their growth habits[5], while categories for the remaining two concepts are grouped using GPT-5 (Singh et al., 2026).

Table 5: Example prompts for each concept

Concept	Prompt	Correct Continuation
ANIMALS	The big cat of genus *Panthera* referred to as the king of the beasts due to its mane and dominant pride structure is called a	lion .
COUNTRIES	The country that contains the Great Pyramid of Giza is	Egypt .
DAYS	The day immediately following the weekend in many calendars is	Monday .
PLANTS	The crop most associated with Popeye gaining strength is	spinach .
VEHICLES	The aircraft that remains airborne through the rotation of large overhead blades is called a	helicopter .
YEARS	The year when the COVID-19 pandemic spread across the world was	2020 .
RANDOM	If several ice cubes are left on a kitchen counter until they completely melt, they become	water .

[5]Source USDA website: https://plants.sc.egov.usda.gov/

Table 6: For each concept, we define a set of categories.

Concept	Category	Entities
ANIMALS	Mammals	beaver, lion, giraffe, horse, koala
	Reptiles	lizard, turtle, dinosaur, crocodile, gecko
	Birds	parrot, owl, duck, eagle, penguin
	Amphibians	frog, salamander, toad, axolotl, newt
	Fish	salmon, shark, pufferfish, clownfish, anglerfish
	Insects	firefly, dragonfly, butterfly, bee, grasshopper
COUNTRIES	Africa	Egypt, Morocco, Kenya, Ethiopia, Madagascar
	Asia	China, India, Japan, Thailand, Vietnam
	Europe	Germany, France, Italy, Spain, Portugal
	North America	Canada, Mexico, Cuba, Jamaica, United States
	South America	Brazil, Argentina, Chile, Colombia, Peru
	Oceania	Australia, Fiji, Samoa, Tonga, New Zealand
VEHICLES	Road	car, bus, truck, bicycle, ambulance
	Rail	train, subway, funicular, monorail, tram
	Air	airplane, helicopter, drone, glider, rocket
	Water	canoe, submarine, yacht, hovercraft, ferry, cruise
PLANTS	Trees	oak, pine, maple, pineapple, banana
	Shrubs	rose, lavender, lilac, boxwood, jasmine
	Herbs	basil, dill, thyme, mint, spinach
	Climbers	grape, passionflower, wisteria, cucumber, watermelon
	Creepers	ivy, pumpkin, strawberry, sweet potato

Concept	Entities
DAYS	Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday
YEARS	1066, 1215, 1347, 1453, 1492, 1517, 1588, 1607, 1620, 1666, 1687, 1776, 1789, 1804, 1815, 1861, 1865, 1903, 1914, 1918, 1929, 1939, 1945, 1963,
RANDOM	water, dog, coffee, bird, king, rain, sky, door, bus, train, window, tree, house, road, book, chair, computer, city, apple, bread

C.2 BEHAVIORAL PERFORMANCE

We now evaluate whether the models can reliably predict the target concept entities using our prompts. For example, for the concept COUNTRIES, a prompt may describe a country such that the expected answer is *United States*. We report the performance of each model on the task of predicting the correct concept word in Tab. 7.

First, we evaluate all four models on the six concept datasets. We report Top-1, Top-5, and continuation accuracy, averaged over all prompts and entities within each concept. Top-1 and Top-5 accuracy measure whether the beginning of the target entity appears among the model's highest-probability next-token predictions, while continuation accuracy measures whether the full target entity appears within the model's greedy continuation of up to five tokens. For example, if the target entity is *United States*, Top-1 is correct if *United* is the highest-probability next token, Top-5 is correct if *United* appears among the five highest-probability next tokens, and continuation accuracy is correct if *United States* appears in the generated continuation. Results are reported in Tab. 7. Overall, all models achieve above 70% Top-5 accuracy across the concept datasets. A high Top-5 accuracy suggests that the hidden activation immediately preceding the concept word generally contains information about that word.

We also measure behavioral performance on the random concept dataset. High scores are not not necessarily a positive or negative signal; rather, high scores are a prerequisite for the validity of applying concept recovery methods to this less coherent set of nouns.

Table 7: **Task performance across concepts and models.** Accuracy measures whether the model predicts the target concept entity for each prompt. *Top-1* reports whether the target entity begins with the model's highest-probability next token, while *Top-5* reports whether it begins with any of the five highest-probability next tokens. This prefix matching accounts for entities that may span multiple tokens. For example, if the target entity is *United States*, a prediction of the token *United* counts as a correct Top-1 match. *Cont* reports whether the complete target entity appears anywhere in the model's greedy continuation of up to five tokens. For example, if the target entity is *Egypt* and the prompt is "The country that contains the Great Pyramid of Giza is", Cont is correct if *Egypt* appears as the continuation. Results are averaged over the entire dataset for each concept.

Model	Metric	Animals	Vehicles	Countries	Days	Plants	Years	Random
Gemma-3-1B	Top-1	0.465	0.458	0.777	0.546	0.403	0.733	0.684
	Top-5	0.707	0.701	0.892	0.922	0.668	0.942	0.892
	Cont	0.444	0.435	0.772	0.524	0.331	0.375	0.704
Gemma-3-27B	Top-1	0.821	0.875	0.920	0.959	0.819	0.948	0.868
	Top-5	0.942	0.958	0.963	0.999	0.962	0.999	0.968
	Cont	0.900	0.872	0.920	0.959	0.834	0.900	0.879
Llama-3.1-8B	Top-1	0.686	0.782	0.890	0.931	0.703	0.885	0.727
	Top-5	0.896	0.931	0.966	1.000	0.918	0.966	0.933
	Cont	0.705	0.786	0.888	0.931	0.649	0.859	0.746
Qwen-3.5-4B	Top-1	0.735	0.744	0.881	0.953	0.638	0.991	0.825
	Top-5	0.905	0.941	0.938	0.999	0.868	1.000	0.958
	Cont	0.764	0.719	0.874	0.942	0.585	0.929	0.829

D EXAMPLES FOR SAMPLING-BASED STEERING OUTPUT

Concept	Method	Can you write a short story where Hildibrand Manderville somehow gets transformed into Gandalf the Grey.	Create a daily itinerary based on the given information. Our family is looking for a 9-day Morocco trip that has light to moderate activity levels.	As a sports commentator, describe the winning play in the final seconds of a championship game.
None	No steering	Hildibrand Manderville awoke to find a long grey beard tangled in his impeccably groomed mustache, a battered pointed hat upon his head, and a gnarled wooden staff clutched in his hand. “By the Twelve!” he cried, as the Warrior of Light and Nashu exchanged bewildered glances.	Day 1: Arrival in Marrakech – Check into your riad, enjoy a leisurely lunch, and take an easy evening stroll through Djemaa el-Fna square to soak in the sights, sounds, and street food. Day 2: Marrakech – Visit the Bahia Palace and Majorelle Garden in the morning,	Five seconds left, down by one – Martinez takes the inbound pass, drives the lane, spins past his defender, and lays it in off the glass as the buzzer sounds! The crowd erupts – they are your champions!
Animals	Prompting	Hildibrand Manderville’s transformation into Gandalf the Grey was a chaotic swirl of mismatched socks and bewildered squirrels, proving that even	Okay, here’s a potential 9-day itinerary for a family trip to Morocco with light to moderate activity levels, incorporating animal experiences and cultural immersion:\n\n**Day 1-2:	With a lightning-fast pass and a desperate heave, the star quarterback lofted the ball intobounds, securing the victory for the Eagles!
	Diff-mean	Owl’s feathers ruffled in the wind as Hildibrand Manderville floated weightlessly amongst the gnarled branches of Elderwood Forest. He was trying – utterly unsuccessfully	African Elephant Safari & Moroccan Culture - A Balanced Itinerary (9 Days)\n\n\nThis itinerary balances wildlife viewing in an African	Hawk! Hawks lead by one with under two minutes left on the clock! The tension is palpable here at Madison Square Garden – you can feel it vibrating
Countries	Prompting	The swirling mists of the Scottish Highlands revealed a chilling shift – Hildibrand Manderville, once a bewildered village idiot, now possessed the	Okay, here’s a possible 9-day itinerary for a family trip to Morocco with light to moderate activity levels	With a desperate heave and a perfect pivot, the star player of Argentina secured the victory, silencing the opposing team’s hopes with a stunning finish!
	Diff-mean	Iceland was unusually warm for November. A persistent drizzle clung to the moss-covered stones as Hildibrand Manderville – or rather, what *was* Hildibrand Manderville – paced	Thailand – A Diverse and Relaxing Escape (Inspired by Moroccan Culture) - 10 Days!\n\nThis itinerary balances cultural experiences with relaxation, offering a blend	Spain! Spain wins it all! The stadium is absolutely *electric* here at Wembley Stadium – you can feel the tension radiating off everyone
Days	Prompting	Hildibrand Manderville’s transformation was a chaotic cascade of Tuesday mornings and misplaced socks, forcing him to confront the unsettling reality	Perhaps Mondays could be for leisurely market explorations, Tuesdays for gentle desert walks,	He caught it just as Sunday surrendered to Monday, securing the win in a moment where every day of the week led to *this* play!
	Diff-mean	Tuesday was always dull for Hildibrand Mandeville. A beige existence consisting primarily of polishing pewter spoons and meticulously arranging dandelion	Wednesday: Arrival in Marrakech - Immerse yourselves! (Moderate, cultural exploration)	Saturday night at Madison Field! The tension is palpable here – you can

Table 8: Comparison of an unsteered outputs, prompting, and Diffmean for the concepts Animals, Countries, and Days. Concept-related words are highlighted in green boxes.

Concept	Method	Can you write a short story where Hildibrand Manderville somehow gets transformed into Gandalf the Grey.	Create a daily itinerary based on the given information. Our family is looking for a 9-day Morocco trip that has light to moderate activity levels.	As a sports commentator, describe the winning play in the final seconds of a championship game.
None	No steering	Hildibrand Manderville awoke to find a long grey beard tangled in his impeccably groomed mustache, a battered pointed hat upon his head, and a gnarled wooden staff clutched in his hand. "By the Twelve!" he cried, as the Warrior of Light and Nashu exchanged bewildered glances.	Day 1: Arrival in Marrakech – Check into your riad, enjoy a leisurely lunch, and take an easy evening stroll through Djemaa el-Fna square to soak in the sights, sounds, and street food. Day 2: Marrakech – Visit the Bahia Palace and Majorelle Garden in the morning,	Five seconds left, down by one – Martinez takes the inbound pass, drives the lane, spins past his defender, and lays it in off the glass as the buzzer sounds! The crowd erupts – they are your champions!
Plants	Prompting	Hildibrand Manderville, lost in a particularly potent patch of petunias, inadvertently triggered a magical shift, transforming him	Okay, here's a possible 9-day itinerary for a family trip to Morocco focused on light to moderate activity levels, incorporating plant life:\n\n**Day 1-3:	Like a Venus flytrap snapping shut on its prey, the wide receiver secured the pass just as the buzzer bloomed, a final, glorious blossom ending the season!
	Diff-mean	Grapevine vines clung thick around Hildibrand's boots as he stumbled through the Whispering Woods. He'd been chasing shadows	Apple please! Here's a possible 9-Day Moroccan Itinerary tailored towards light to moderate activities, blending culture and relaxation	Maplewood – The stadium is absolutely buzzing! Ten seconds left on the clock, 18 points to go… and it's *this*!
Vehicles	Prompting	Imagine a sleek sports car suddenly morphing into a lumbering steam locomotive – that's essentially Hildibrand's predicament as, after a particularly flamboyant	Like planning a cross-country road trip in a comfortable RV—balancing sightseeing with relaxing drives—a nine-day Moroccan adventure could begin with exploring Marrakech via	He dodged three defenders like expertly navigating rush hour traffic in a compact car, then launched a full-court shot—a
	Diff-mean	Airship Nimbus drifted lazily over the bruised purple skies of Middle-earth. Inside, Hildibrand Mandeville – formerly renowned for meticulously arranging	Space Rover Shuttle - The first day of your Moroccan adventure! Let's build a relaxed and enjoyable 9-Day Marocitinerary,	airplane airplane! Jets fly! Folks, you are NOT going to believe what just happened here at Madison Square Garden tonight!
Years	Prompting	The relentless march of years, from nineteen seventy-three to now, has slowly eroded the foundations of Hildibrand's carefully constructed world, forcing him to confront	Planning nine years' worth of Moroccan adventures into just nine days requires carefully allocating each of those precious	For seventy-two years fans had waited, but in the final three seconds of 2024, Johnson soared—a moment etched forever in the
	Diff-mean	1987 was not a good year for Hildibrand Manderville. He'd been obsessively collecting bottle caps in Mrs. Higgins' garden	17th August, 2024 - Today (Day 1) – Arrival in Marrakech & Medina Exploration!\n\n* **Morning:** Arrive at Marrakech Menara Airport (RAK).	2018 – October 31st - The stadium was electric! A nail-biting finish to the NFL Championship Game between the Eagles and the Browns here at SoFi Stadium!

Table 9: Continuation of Table 8 for the concepts Plants, Vehicles, and Years. Concept-related words are highlighted in green boxes.

E LLM-JUDGE

E.1 EVALUATING AGREEMENT BETWEEN LLM JUDGE AND HUMAN

We evaluate agreement between our LLM judge and human annotation. For each of 7 concepts, we randomly sample outputs from all methods and for 4 models. Human annotator scores for concept, fluency and instruct scores (scale 0-2). We use gpt-5.6-luna as the LLM judge. We report agreement per score using ordinal-weighted Krippendorff's α.

Table 10: Krippendorff's α between human and judge model. We observed best alignment with gpt-5.6-luna and use this judge for all our evaluations.

Judge model	**Concept**	**Fluency**	**Instruction**	n
gpt-5.6-luna	$+0.7995$	$+0.6989$	$+0.8005$	280

E.2 LLM JUDGE PROMPTS

We use the GPT-5.6-luna Singh et al. (2026) as our LLM judge. We use concept-specific prompts to evaluate the concept score, while the same prompts are used across concepts for fluency and instruction following. We additionally use an extraction prompt to identify all entities mentioned in each generation and assign their corresponding taxonomies. Finally, we use a seen–unseen classification prompt to determine whether each extracted entity appeared in the concept dataset used for localization or is new.

Judge Prompt — Concept: Animals

Judge whether OUTPUT contains the animal concept.

Scores:
2 = The output contains an animal-related word used with its intended semantic meaning: referring literally to an animal, animals, animal categories. Examples: "The wolves hunted at dusk," "a bird flew overhead,"
1 = An animal-related word or recognizable substring appears, but it is not used in the literal animal sense. This includes brand names, vehicles, sports teams, mascots, place names, surnames, or metaphors. Examples: "the Wolves won," "the Raptors scored," "He drives a Jaguar," "seal the envelope."
0 = No animal-related word, synonym, category term, or recognizable substring appears anywhere in the output.

Examples:

OUTPUT: The eagle circled above the mountain.
{"score":2,"reason":"The word 'eagle' directly refers to an animal."}

OUTPUT: He bought a used Jaguar last week.
{"score":1,"reason":"The animal-related word 'Jaguar' is used as a car brand."}

OUTPUT: The meeting starts at noon.
{"score":0,"reason":"No animal-related word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Concept: Vehicles

Judge whether OUTPUT contains the vehicle concept.

Scores:
2 = The vehicle concept is present and used with its intended transportation-related meaning. This includes specific vehicles and generic vehicle-related terms, such as "car," "truck," "bus," "van," "bicycle," "bike," "motorcycle," "train," "plane," "airplane," "jet," "helicopter," "drone," "boat," "ship," "vehicle," "transportation," "transit," "traffic," "road," "highway," "street," "lane," "commute," "ride," and "driving."
1 = A vehicle-related word or recognizable substring appears, but it is not used with its intended vehicle or transportation meaning. This includes non-vehicle meanings, idioms, brands, model names, company names, sports teams, mascots, proper names, or metaphors. Examples include "train" meaning to practice, "plane" meaning a flat surface.
0 = No vehicle-related word, synonym, category term, or recognizable substring appears anywhere in the OUTPUT.

Examples:

OUTPUT: The bus arrived at the station on time.
{"score":2,"reason":"The word 'bus' directly refers to a vehicle."}

OUTPUT: I need to train my new employee this week.
{"score":1,"reason":"'train' is used to mean teach, not the vehicle."}

OUTPUT: The weather is nice today.
{"score":0,"reason":"No vehicle-related word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Concept: Days of the Week

Judge whether OUTPUT contains the day-of-week concept.

Scores:
2 = The day-of-week concept is present and used with its intended temporal meaning: referring to a day or group of days in the weekly calendar. This includes "Monday," "Tuesday," "Wednesday," "Thursday," "Friday," "Saturday," "Sunday," "weekday," "weekdays," "weekend," "weekends," "day of the week," and similar expressions.
1 = A day-of-week word or recognizable substring appears, but it is not used with its intended calendar meaning. This includes names of people, pets, characters, brands, titles, places, organizations, or other proper names, as well as idioms or unrelated expressions. Examples include "Wednesday" as a person or character's name, "Friday" as a pet's name or "Sunday" as a restaurant or business name. Assign 1 whenever the term appears but the context does not refer to the actual day or days of the week.
0 = No day-of-week-related word, synonym, category term, or recognizable substring appears anywhere in the OUTPUT.

Examples:

OUTPUT: Let's meet on Wednesday for lunch.
{"score":2,"reason":"'Wednesday' directly refers to a day of the week."}

OUTPUT: Friday, my dog is very excited.
{"score":1,"reason":"'Friday' is used as a pet's name, not the day."}

OUTPUT: The store closes at 9pm.
{"score":0,"reason":"No day-of-week word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Concept: Years

Judge whether OUTPUT contains the year concept.

Scores:
2 = The year concept is present and used with its intended calendar or historical-period meaning. This includes specific calendar years such as "2024," "1999," or "1776"; decades such as "1980s" or "the '90s"; centuries such as "19th century"; and generic terms such as "year," "years," "annual," "yearly," and similar expressions referring to calendar years or historical periods.
1 = A year-related number or recognizable substring appears, but it is not used with its intended calendar or historical-period meaning. This includes model numbers, room numbers, prices, quantities, scores, identification numbers, measurements, addresses, telephone numbers, product codes, or other numeric values that happen to resemble a year. Assign 1 when a year-like number appears without sufficient context to indicate a calendar year, unless the surrounding context clearly establishes its temporal or historical meaning.
0 = No year-related word, calendar reference, historical-period term, or recognizable year-like number appears anywhere in the OUTPUT.

Examples:

OUTPUT: The revolution began in 1789.
{"score":2,"reason":"'1789' directly refers to a specific year."}

OUTPUT: I bought seat number 1969 for the concert.
{"score":1,"reason":"'1969' is a seat number, not a year reference."}

OUTPUT: The cat sat on the mat.
{"score":0,"reason":"No year-related word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Concept: Plants

Judge whether OUTPUT contains the plant concept.

Scores:
2 = The plant concept is present and used with its intended botanical meaning. This includes specific plants and generic plant-related terms such as "rose," "oak," "pine," "fern," "bamboo," "cactus," "flower," "tree," "grass," "shrub," "vine," "herb," "moss," "plant," "plants," "vegetation," "flora," "foliage," "botanical," "leaf," "leaves," "seed," "roots," "stem," "garden," "forest," and similar terms referring to plants or plant parts.
1 = A plant-related word or recognizable substring appears, but it is not used with its intended botanical meaning. This includes personal names, brands, products, places, organizations, idioms, technical terms, or metaphors. Examples include "Rose" or "Basil" as a person's name, "mint" meaning to produce coins or something new, "root" in mathematics or computing, "plant" meaning a factory or to place something. Assign 1 whenever the term appears but the context clearly does not refer to an actual plant, plant part, or botanical setting.
0 = No plant-related word, synonym, category term, or recognizable substring appears anywhere

in the OUTPUT.

Examples:

OUTPUT: The oak tree provided shade.
{"score":2,"reason":"'oak tree' directly refers to a plant."}

OUTPUT: Rose finished her homework early.
{"score":1,"reason":"'Rose' is used as a person's name, not the plant."}

OUTPUT: The car needs an oil change.
{"score":0,"reason":"No plant-related word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Concept: Countries

Judge whether OUTPUT contains the country concept.

Scores:
2 = The country concept is present and used with its intended geographic, national, or nationhood-related meaning. This includes specific country names such as "France," "Brazil," "Japan," and "Nigeria"; demonyms such as "French," "Brazilian," "Japanese," and "Nigerian"; and generic terms such as "country," "countries," "nation," "nations," "state," "states," "national," "international," "foreign," "border," "homeland," and similar expressions referring to countries or nationhood.
1 = A country-related word or recognizable substring appears, but it is not used with its intended country meaning. This includes homonyms, personal names, brands, places that are not countries, objects or other unrelated meanings. Examples include "turkey" meaning the bird, "China" meaning dishware, "Chad" as a person's name. Assign 1 whenever the term appears but the context clearly does not refer to a country, nation, nationality, or international border.
0 = No country-related word, synonym, category term, or recognizable substring appears anywhere in the OUTPUT.

Examples:

OUTPUT: She recently traveled to Japan.
{"score":2,"reason":"'Japan' directly refers to a country."}

OUTPUT: We ate turkey for dinner.
{"score":1,"reason":"'turkey' refers to the bird/food, not the country."}

OUTPUT: The meeting starts at noon.
{"score":0,"reason":"No country-related word or concept appears in the output."}

Return only:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Instruction Following

Evaluate ONLY whether the OUTPUT follows, attempts, or plausibly begins to follow the INSTRUCTION.

Ignore fluency, grammar, factual correctness, logical correctness, and completeness.

Scores:

2 — FOLLOWS / CLEARLY ATTEMPTS
The OUTPUT is meaningfully on the requested task and could plausibly be part of a response fulfilling it.

Give 2 even if:

- it is only setup, background, an introduction, or a clarifying question directly in service of the task;
- required details or later parts have not appeared yet;
- it is truncated;
- names, facts, ingredients, dates, places, or other content details are wrong;
- it misdescribes or misidentifies the requested subject while still clearly attempting the requested kind of answer.

For stories, commentary, itineraries, recipes, etc., beginning the requested type of response is enough. Do not require the key event, all requested characters, every day, ingredients, or final result to appear yet.

1 — RELATED BUT CLEARLY DEVIATES
The OUTPUT has meaningful overlap with the instruction, but what is already visible clearly prevents it from satisfying an important requirement.

Use 1 when:

- it explicitly switches to a clearly different task or deliverable;
- it attempts the requested structural/formatting task but visibly violates a required format that later continuation cannot fix;
- it performs a related task but not the requested one.

Do NOT use 1 merely because details are incorrect or because the requested answer has not progressed far enough yet.

0 — NOT FOLLOWING
The OUTPUT makes no meaningful attempt at the requested task. It is unrelated, bare junk, or performs an entirely different task with no useful connection.

Important:

- Judge the response's TASK TRAJECTORY, not its factual accuracy.
- Ask: "Could this reasonably be the beginning or preparation for the requested response?" If yes, usually score 2.
- Wrong content inside the requested type of response is usually 2, not 1.
- An irrelevant prefix does not lower the score if the remaining output clearly attempts the instruction.
- Never penalize truncation.
- For hard formatting requirements, judge what is already visible: a visible unrecoverable formatting failure is 1.
- Use 0 only when there is no meaningful attempt; do not use 0 merely for a bad or incorrect attempt.

Return only valid JSON:
{"reason":"one short sentence reason for the score","score":0|1|2}

Judge Prompt — Fluency

Evaluate ONLY the fluency and readability of the OUTPUT.

Ignore completeness/truncation, relevance, factual correctness, and technical correctness of code/Markdown.

Scores:

2 — FLUENT
Natural, readable English or readable structured content. Minor typos, grammar/punctuation issues, spacing mistakes, or truncation are allowed.

1 — LOCAL DISFLUENCY
Mostly fluent, with ONE limited defect such as:

- one malformed/fused word;
- one stray symbol, word, or short disconnected phrase;
- a brief unexplained language switch.

A short local defect does not make the whole output severely garbled.

0 — SEVERELY DISFLUENT
Use 0 if ANY of these apply:

- the beginning contains more than a single local garbling/corruption, even if later text is fluent;
- multiple consecutive tokens are gibberish, corrupted, encoded, malformed, or machine-like;
- 3+ disconnected entities/keywords of the same type are listed or fused together (countries, animals, weekdays, vehicles, names, etc.) without normal sentence structure;
- pathological repetition or extensive missing spaces/fused words occurs;
- disconnected/garbled/list-like material occupies roughly 1/3 or more of the output;
- a substantial section is not natural prose or readable structured content.

IMPORTANT:

- Evaluate 0 conditions BEFORE deciding between 1 and 2.
- Later fluent prose does NOT rescue a severe garbled opening.
- Do NOT reinterpret garbage, entity dumps, or repetition as intentional literary style.
- One malformed word $\rightarrow$ usually 1.
- Multiple malformed/gibberish tokens or 3+ disconnected category items $\rightarrow$ 0.
- A long entity/keyword run $\rightarrow$ 0 even if it is less than 1/3 when it clearly disrupts the beginning.
- Do NOT penalize truncation itself.
- For code/Markdown, ignore syntax/fence correctness; judge readability.

Return only valid JSON:
{"reason":"one short sentence reason for the score","score":0|1|2}

Diversity Prompt — Concept: Animals

Read the following sentence and extract all animal mentions.
Sentence: {sentence}

Rules:

- Extract every animal word, even if used in a non-animal context (e.g. "Python" in programming → snake, "Jaguar" as a car → big cat)
- Normalize each name to its canonical base/singular form (e.g. "lions" → "lion", "fisher" → "fish", "birdy" → "bird"). Always use the exact same canonical name for every mention of the same animal.
- Breed-level mentions are distinct (e.g. "labrador" is not collapsed to "dog")
- Generic terms count too ("creature", "beast", "mammal", "bird")
- For each animal, assign a taxonomy from EXACTLY this list: {taxonomy_options}. If the animal does not clearly belong to any taxonomy in that list, assign "Other".
- If no animals are present, return empty list

Return ONLY this JSON:

```
{
  "items": [
    {"name": "<canonical animal name>", "taxonomy": "<one of: {taxonomy_options}, or Other>"},
    ...
  ]
}
```

Diversity Prompt — Concept: Vehicles

Read the following sentence and extract all vehicle mentions.
Sentence: {sentence}

Rules:

- Extract every vehicle word, even if used metaphorically (e.g. "drive the meeting" → car/generic, "train your model" → train)
- Normalize each name to its canonical base/singular form (e.g. "cars" → "car", "buses" → "bus"). Always use the exact same canonical name for every mention of the same vehicle.
- Model-level mentions are distinct (e.g. "Tesla Model S" is not collapsed to "car")
- Generic terms count too ("vehicle", "transport", "ride")
- For each vehicle, assign a taxonomy from EXACTLY this list: {taxonomy_options}. If the vehicle does not clearly belong to any taxonomy in that list, assign "Other".
- If no vehicles are present, return empty list

Return ONLY this JSON:

```
{
  "items": [
    {"name": "<canonical vehicle name>", "taxonomy": "<one of: {taxonomy_options}, or Other>"},
    ...
  ]
}
```

Diversity Prompt — Concept: Days of the Week

Read the following sentence and extract all day-of-week mentions.
Sentence: {sentence}

Rules:

- Extract every day-of-week word (Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday)
- Also extract indirect references like "weekday", "weekend", "daily", "someday"
- Normalize each name to its canonical form (e.g. capitalized day name, or lowercase for generic references like "weekday"). Always use the exact same canonical name for every mention of the same day/reference.
- If no day references are present, return empty list

Return ONLY this JSON:
{
"items": [day1, day2,..]
}

Diversity Prompt — Concept: Years

Read the following sentence and extract all year mentions.
Sentence: {sentence}

Rules:

- Extract every explicit 4-digit year (e.g. 1066, 1776, 1945, 2020)
- Ignore numbers that are not years (ages, quantities, page numbers, model numbers, IDs, etc.)
- Preserve the year exactly as written as the name
- If no years are present, return an empty list

Return ONLY this JSON:
{
"items": [year1, year2,..]
}

Diversity Prompt — Concept: Plants

Read the following sentence and extract all plant mentions.
Sentence: {sentence}

Rules:

- Extract every plant word, even if used metaphorically (e.g. "she blossomed" → blossom/flower, "rooted in tradition" → root/generic)
- Normalize each name to its canonical base/singular form (e.g. "roses" → "rose"). Always use the exact same canonical name for every mention of the same plant.
- Species-level mentions are distinct (e.g. "oak" is not collapsed to "tree")
- Generic terms count too ("plant", "vegetation", "flora", "foliage")
- For each plant, assign a taxonomy from EXACTLY this list: {taxonomy_options}. If the plant does not clearly belong to any taxonomy in that list, assign "Other".
- If no plants are present, return empty list

Return ONLY this JSON:
{
"items": [
{"name": "<canonical plant name>", "taxonomy": "<one of: {taxonomy_options}, or Other>"},
...
]
}

Diversity Prompt — Concept: Countries

Read the following sentence and extract all country or nationality mentions.
Sentence: {sentence}

Rules:

- Extract every country name (e.g. "France", "Brazil") and demonym (e.g. "French", "Brazilian", "Japanese")
- Also extract vague references like "overseas", "foreign", "internationally" as generic
- Normalize each name to its canonical country name (e.g. "French" → "France", "Brazilian" → "Brazil"). Always use the exact same canonical name for every mention of the same country.
- For each mention, assign a taxonomy from EXACTLY this list: {taxonomy_options}. If the country does not clearly belong to any region in that list, assign "Other".
- If no country references are present, return empty list

Return ONLY this JSON:

```
{
  "items": [
    {"name": "<canonical country name or generic reference>", "taxonomy": "<one of: {taxonomy_options}, or Other>"},
    ...
  ]
}
```

Unseen Extraction Prompt

You are performing Out-of-Distribution (OOD) detection for the {concept} concept.

Known in-distribution {concept} instances:
{training_str}

Extracted {concept} instances:
{instances_str}

Task:
Identify OOD {concept} instances.

Rules for instances:

- An instance is OOD if it does NOT match any known {concept} instance.
- The provided {concept} instance list is the complete allowed vocabulary. Do not use outside knowledge to consider a {concept} instance as in-distribution.
- Matching is case-insensitive.
- Allow basic morphological variations.
- Do NOT match synonyms or related concepts.
- Ignore generic terms that are not specific instances of the concept.
- Preserve the original extracted instance name.

Return ONLY this JSON:

```
{
  "ood_instances": [
    "<instance_name>",
    ...
  ]
}
```

F SUPPLEMENTARY EXPERIMENTS

F.1 METRIC BREAKDOWN FOR MAXBENCH SCORE

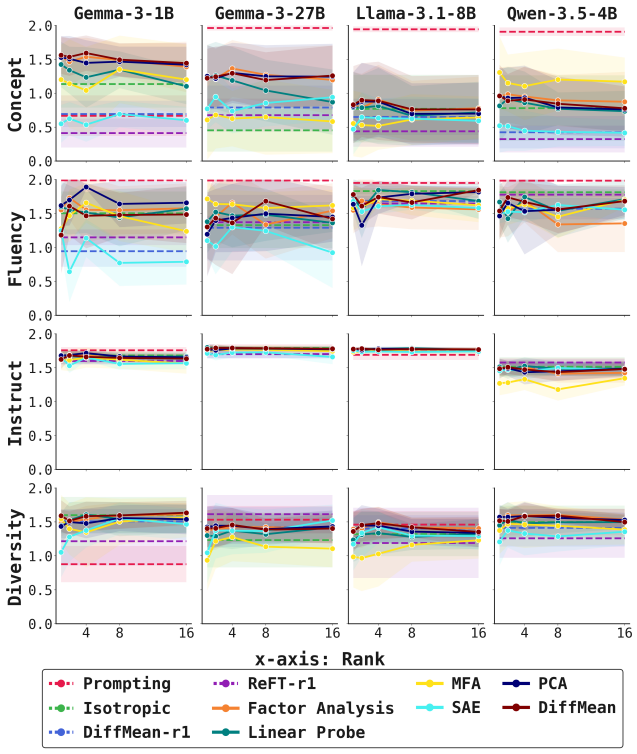


Figure 7: All models metric breakdown averaged across concepts.

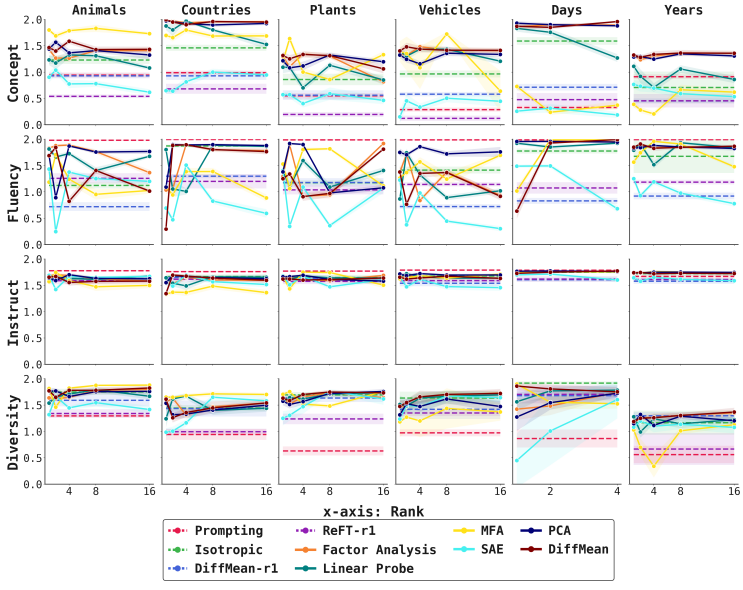


Figure 8: Gemma-3-1B metric breakdown averaged across concepts.

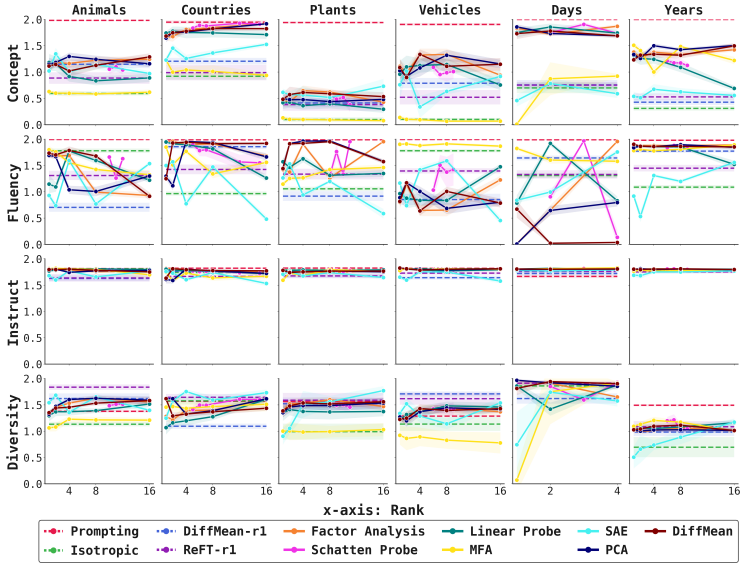


Figure 9: Gemma-3-27B metric breakdown averaged across concepts.

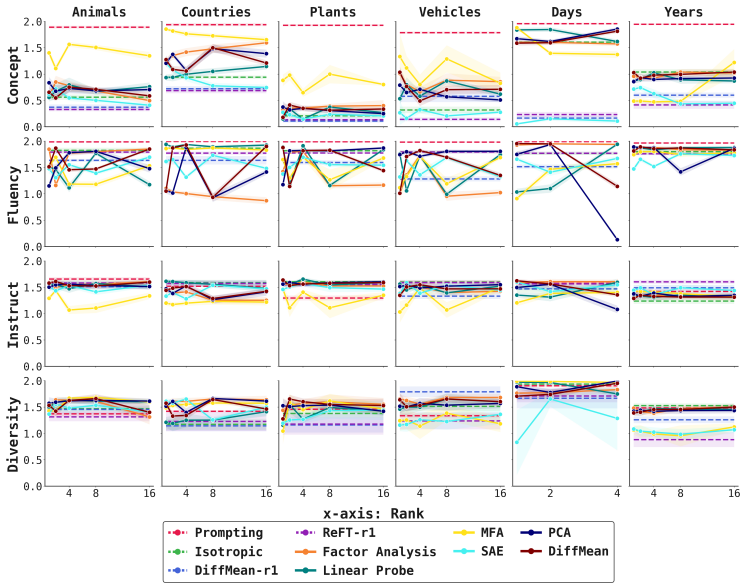


Figure 10: Qwen-3.5-4B metric breakdown averaged across concepts.

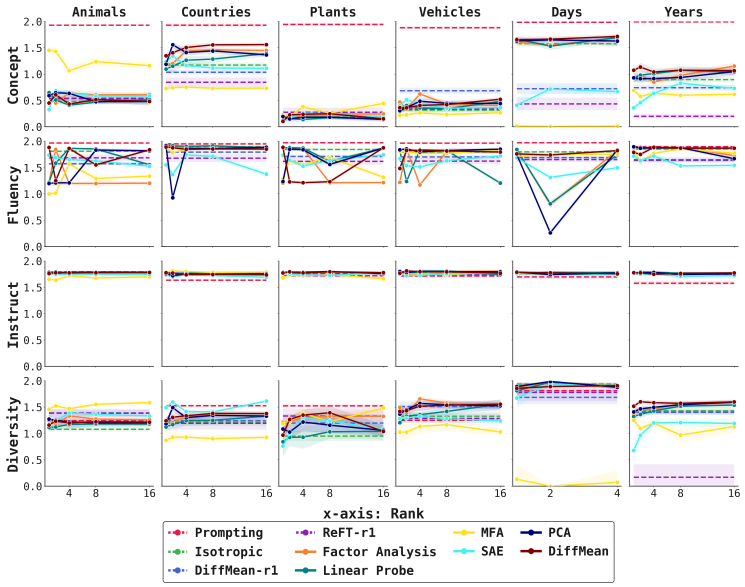


Figure 11: Llama-3.1-8B metric breakdown averaged across concepts.

F.2 METRIC META-EVALUATION

We now ask whether each of the metrics in the MAXBENCH score empirically acts an independent factor, or whether there exist correlations between metrics (such that the number of metrics to measure could be condensed). To investigate, we compute Pearson and Spearman correlations across all five metrics to assess the degree to which they covary either linearly or in their relative ranking of methods, respectively.

We observe that there is a notable correlation between concept scores and both the instance and category-level diversity scores. Instruct score has somewhat negative correlation to concept score and fluency score is mostly neutral. We report both pearson and spearman correlation in Fig. 12.

Does each metric (including the proposed diversity score) measure truly independent factors of performance, or do some factors strongly covary? We find (Fig. 12) that the concept score is somewhat correlated with instance and category diversity. This is expected, as the diversity score can only be maximized when all generated outputs contain concept-related values. The concept score and instruction score are negatively correlated, suggesting that when the target concept is injected more strongly, it can become harder for the model to stay on topic with respect to the original instruction. Fluency scores have relatively weak correlations with other metrics.

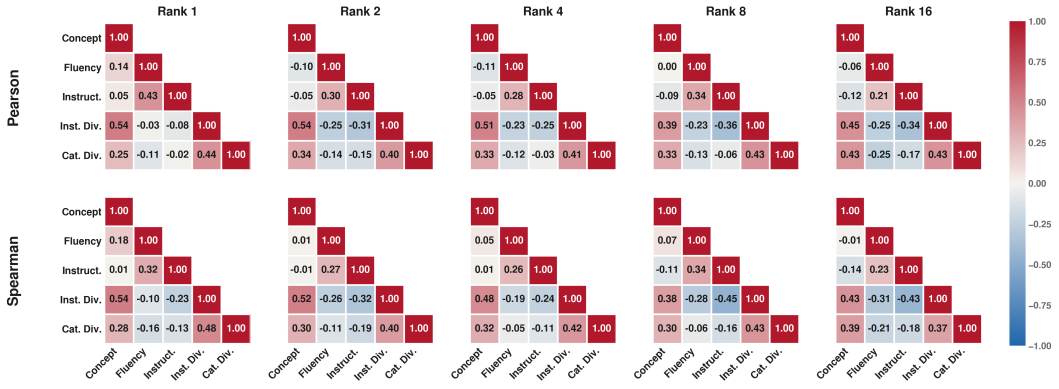


Figure 12: Pearson and Spearman correlation heatmaps for all the metrics.

F.3 AFFINE VS LINEAR COMPARISON

We observe similar trends for affine vs linear offset for other models as we did for Gemma-3-1B.

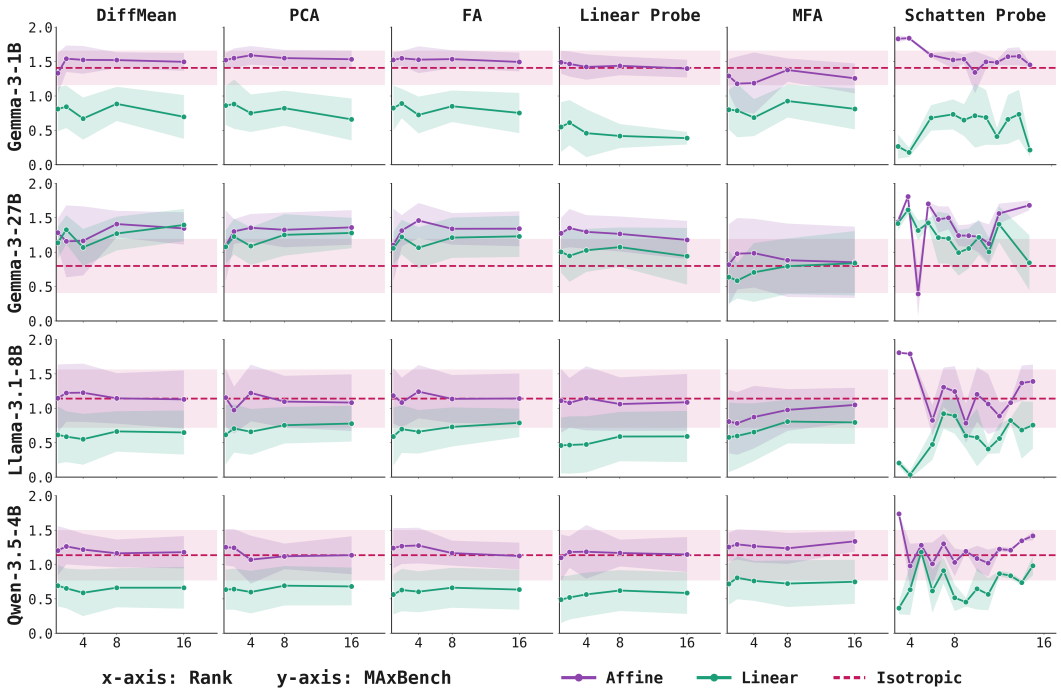


Figure 13: Comparison of multinomial concept localization methods for all models with and without an offset (affine and linear, respectively). The x-axis shows the rank hyperparameter for each method, and the y-axis reports the MAXBENCH score. For the Schatten probe, we plot the effective rank of the learned weight matrix as the Schatten p-norm parameter is varied from 0.1 to 0.9. For each method, we average the scores across all concepts and compute std. dev. via different data splits. Sampling around the concept mean offset is consistently significantly better than assuming the subspace is centered at the origin.

F.4 SAMPLING EVALUATION RESULTS ACROSS MODELS

Here we discuss additional results for different models families. We have results for breakdown of different metrics in Tab 11. We show instance diversity and category diversity separately to indicate how much each component contributes to overall diversity.[6] Results across ranks are in Fig. 14 for all models.

Embedding baseline. We also include a embedding baseline to show the performance of different methods when no context is provided. Representations are extracted directly from the embeddings of the concept's instance tokens. This allows us to evaluate how much contextual information contributes to recovering a concept representation. Across models, we observe better performance when using representations from intermediate layers. This suggests that steering is operating over a space that contains more than the concept values' token identities.

[6]Instance and category diversity do not always sum directly to the reported Diversity Score. This is because concepts without a category structure, such as DAYS and YEARS, the Diversity Score is defined as twice the instance diversity.

Table 11: Scores on the test set for Gemma-3-1B, Gemma-3-27B, Llama-3.1-8B, and Qwen3.5-4B. We report *Concept*, *Fluency*, *Instruction*, and *Diversity* scores, together with the composite AXBENCH and MAXBENCH scores. Full metric definitions are given in §3.3. For each concept and method, we select the rank that achieves the highest mean MAXBENCH score. Each reported value is the mean ± standard deviation over all the data splits.

	Method	Concept	Fluency	Instruction	Inst. Div.	Cat. Div.	Diversity	AXBENCH	MAXBENCH
Gemma-3-1B	Prompting	0.67 ± 0.30	**1.99** ± 0.01	**1.76** ± 0.04	0.35 ± 0.10	0.58 ± 0.21	0.84 ± 0.28	1.11 ± 0.33	0.99 ± 0.27
	Isotropic	1.14 ± 0.35	1.51 ± 0.32	1.68 ± 0.06	0.76 ± 0.14	0.87 ± 0.06	1.59 ± 0.25	1.36 ± 0.27	1.41 ± 0.25
	Embedding	0.87 ± 0.30	1.40 ± 0.43	1.69 ± 0.08	0.58 ± 0.19	0.86 ± 0.06	1.33 ± 0.41	1.16 ± 0.31	1.17 ± 0.34
	DiffMean-r1	0.70 ± 0.19	0.95 ± 0.23	1.61 ± 0.06	0.73 ± 0.11	0.80 ± 0.09	1.51 ± 0.18	0.93 ± 0.16	1.03 ± 0.14
	ReFT-r1	0.41 ± 0.21	1.15 ± 0.12	1.60 ± 0.05	0.54 ± 0.19	0.72 ± 0.10	1.22 ± 0.36	0.72 ± 0.28	0.76 ± 0.28
	MFA	1.28 ± 0.46	1.47 ± 0.30	1.67 ± 0.10	0.77 ± 0.17	0.78 ± 0.15	1.52 ± 0.31	1.39 ± 0.24	1.40 ± 0.21
	SAE (top-k)	0.62 ± 0.24	1.35 ± 0.19	1.67 ± 0.05	0.52 ± 0.18	0.71 ± 0.10	1.13 ± 0.35	0.97 ± 0.25	0.97 ± 0.29
	Factor Analysis	1.51 ± 0.32	1.83 ± 0.08	1.69 ± 0.06	0.71 ± 0.07	0.86 ± 0.06	1.53 ± 0.12	1.65 ± 0.13	1.61 ± 0.11
	Schatten Probe	1.46 ± 0.32	1.83 ± 0.08	1.70 ± 0.05	0.75 ± 0.08	0.88 ± 0.05	1.60 ± 0.13	1.63 ± 0.15	1.62 ± 0.12
	Linear Probe	1.40 ± 0.29	1.73 ± 0.17	1.69 ± 0.06	0.71 ± 0.10	0.85 ± 0.07	1.52 ± 0.17	1.58 ± 0.17	1.56 ± 0.14
	PCA	1.50 ± 0.30	1.86 ± 0.07	1.70 ± 0.05	0.72 ± 0.07	0.87 ± 0.06	1.54 ± 0.14	**1.66** ± 0.13	1.62 ± 0.11
	Diff Mean (Pairs)	**1.54** ± 0.33	1.75 ± 0.20	1.68 ± 0.07	**0.78** ± 0.08	**0.89** ± 0.04	**1.63** ± 0.15	1.63 ± 0.17	**1.62** ± 0.12
Gemma-3-27B	Prompting	**1.96** ± 0.04	**1.99** ± 0.01	1.78 ± 0.06	0.78 ± 0.09	0.71 ± 0.13	1.54 ± 0.20	**1.91** ± 0.02	**1.79** ± 0.06
	Isotropic	0.46 ± 0.31	1.33 ± 0.34	1.79 ± 0.03	0.57 ± 0.20	0.67 ± 0.15	1.22 ± 0.39	0.74 ± 0.38	0.80 ± 0.39
	Embedding	0.98 ± 0.34	1.04 ± 0.49	1.73 ± 0.06	0.58 ± 0.21	0.87 ± 0.05	1.31 ± 0.46	1.11 ± 0.38	1.13 ± 0.42
	DiffMean-r1	0.79 ± 0.32	1.29 ± 0.48	1.70 ± 0.07	0.68 ± 0.15	0.80 ± 0.12	1.42 ± 0.29	1.06 ± 0.27	1.10 ± 0.21
	ReFT-r1	0.68 ± 0.24	1.37 ± 0.09	1.70 ± 0.06	**0.80** ± 0.15	0.84 ± 0.10	**1.60** ± 0.27	1.04 ± 0.21	1.13 ± 0.21
	MFA	0.76 ± 0.52	1.68 ± 0.26	1.76 ± 0.08	0.57 ± 0.18	0.63 ± 0.15	1.24 ± 0.32	1.06 ± 0.54	1.06 ± 0.49
	SAE (top-k)	0.87 ± 0.34	1.46 ± 0.34	1.74 ± 0.07	0.67 ± 0.10	0.72 ± 0.15	1.38 ± 0.22	1.17 ± 0.23	1.21 ± 0.21
	Factor Analysis	1.33 ± 0.48	1.73 ± 0.25	1.78 ± 0.04	0.64 ± 0.11	**0.87** ± 0.06	1.44 ± 0.21	1.53 ± 0.28	1.49 ± 0.22
	Schatten Probe	1.26 ± 0.50	1.68 ± 0.24	1.79 ± 0.03	0.65 ± 0.08	0.86 ± 0.06	1.44 ± 0.13	1.49 ± 0.32	1.46 ± 0.25
	Linear Probe	1.18 ± 0.53	1.72 ± 0.19	**1.79** ± 0.03	0.58 ± 0.07	0.81 ± 0.07	1.31 ± 0.16	1.44 ± 0.37	1.37 ± 0.27
	PCA	1.31 ± 0.48	1.49 ± 0.44	1.78 ± 0.04	0.65 ± 0.13	0.86 ± 0.04	1.45 ± 0.23	1.39 ± 0.30	1.38 ± 0.22
	Diff Mean (Pairs)	1.29 ± 0.42	1.51 ± 0.50	1.79 ± 0.03	0.65 ± 0.11	0.86 ± 0.06	1.45 ± 0.20	1.39 ± 0.28	1.39 ± 0.20
Llama-3.1-8B	Prompting	**1.94** ± 0.04	**1.95** ± 0.03	1.69 ± 0.07	**0.75** ± 0.08	0.66 ± 0.16	1.46 ± 0.19	**1.85** ± 0.03	**1.73** ± 0.07
	Isotropic	0.77 ± 0.50	1.83 ± 0.04	1.76 ± 0.03	0.60 ± 0.19	0.66 ± 0.15	1.32 ± 0.33	1.12 ± 0.46	1.14 ± 0.42
	Embedding	0.27 ± 0.23	1.68 ± 0.04	1.77 ± 0.02	0.30 ± 0.17	0.58 ± 0.18	0.72 ± 0.44	0.54 ± 0.36	0.50 ± 0.39
	DiffMean-r1	0.65 ± 0.27	1.68 ± 0.07	1.75 ± 0.02	0.64 ± 0.13	0.77 ± 0.11	1.41 ± 0.20	1.06 ± 0.32	1.12 ± 0.29
	ReFT-r1	0.44 ± 0.23	1.65 ± 0.06	1.77 ± 0.02	0.51 ± 0.25	0.78 ± 0.07	1.19 ± 0.52	0.82 ± 0.28	0.83 ± 0.37
	MFA	0.57 ± 0.40	1.63 ± 0.24	1.74 ± 0.06	0.52 ± 0.27	0.68 ± 0.07	1.10 ± 0.45	0.88 ± 0.46	0.90 ± 0.45
	SAE (top-k)	0.72 ± 0.34	1.49 ± 0.13	1.73 ± 0.03	0.68 ± 0.15	0.73 ± 0.08	1.43 ± 0.26	1.07 ± 0.27	1.13 ± 0.25
	Factor Analysis	0.97 ± 0.51	1.50 ± 0.31	1.76 ± 0.03	0.70 ± 0.14	**0.80** ± 0.11	**1.52** ± 0.21	1.24 ± 0.44	1.28 ± 0.39
	Schatten Probe	0.99 ± 0.55	1.59 ± 0.30	1.76 ± 0.03	0.69 ± 0.14	0.80 ± 0.10	1.51 ± 0.20	1.27 ± 0.43	1.30 ± 0.38
	Linear Probe	0.91 ± 0.53	1.61 ± 0.29	1.77 ± 0.03	0.62 ± 0.17	0.73 ± 0.15	1.39 ± 0.28	1.20 ± 0.48	1.22 ± 0.43
	PCA	0.91 ± 0.52	1.60 ± 0.28	**1.77** ± 0.03	0.65 ± 0.16	0.76 ± 0.14	1.45 ± 0.26	1.21 ± 0.45	1.24 ± 0.41
	Diff Mean (Pairs)	0.97 ± 0.55	1.61 ± 0.27	1.77 ± 0.03	0.68 ± 0.15	0.80 ± 0.10	1.50 ± 0.22	1.25 ± 0.45	1.29 ± 0.40
Qwen3.5-4B	Prompting	**1.91** ± 0.06	**1.99** ± 0.01	1.51 ± 0.13	0.77 ± 0.10	0.67 ± 0.10	1.50 ± 0.20	**1.77** ± 0.06	**1.69** ± 0.08
	Isotropic	0.78 ± 0.48	1.81 ± 0.04	1.52 ± 0.14	0.69 ± 0.16	0.79 ± 0.08	1.50 ± 0.24	1.08 ± 0.39	1.14 ± 0.37
	Embedding	0.69 ± 0.25	1.53 ± 0.20	1.44 ± 0.10	0.58 ± 0.16	0.83 ± 0.07	1.34 ± 0.29	1.04 ± 0.21	1.09 ± 0.20
	DiffMean-r1	0.42 ± 0.24	1.57 ± 0.16	1.49 ± 0.09	0.66 ± 0.18	0.77 ± 0.12	1.42 ± 0.28	0.75 ± 0.32	0.83 ± 0.31
	ReFT-r1	0.32 ± 0.20	1.78 ± 0.03	**1.58** ± 0.05	0.57 ± 0.16	0.71 ± 0.10	1.26 ± 0.29	0.65 ± 0.29	0.72 ± 0.27
	MFA	1.29 ± 0.31	1.61 ± 0.23	1.29 ± 0.13	**0.79** ± 0.14	0.70 ± 0.10	1.51 ± 0.26	1.35 ± 0.16	1.38 ± 0.16
	SAE (top-k)	0.57 ± 0.37	1.46 ± 0.14	1.42 ± 0.08	0.72 ± 0.16	0.67 ± 0.09	1.39 ± 0.26	0.86 ± 0.35	0.93 ± 0.33
	Factor Analysis	0.99 ± 0.42	1.62 ± 0.33	1.49 ± 0.11	0.77 ± 0.08	0.86 ± 0.06	**1.61** ± 0.11	1.22 ± 0.28	1.29 ± 0.25
	Schatten Probe	0.94 ± 0.43	1.75 ± 0.21	1.52 ± 0.08	0.72 ± 0.10	0.87 ± 0.05	1.57 ± 0.12	1.24 ± 0.32	1.30 ± 0.28
	Linear Probe	0.91 ± 0.42	1.73 ± 0.27	1.50 ± 0.11	0.71 ± 0.09	0.86 ± 0.07	1.55 ± 0.12	1.23 ± 0.32	1.28 ± 0.28
	PCA	0.97 ± 0.43	1.76 ± 0.17	1.49 ± 0.10	0.73 ± 0.08	0.86 ± 0.06	1.57 ± 0.11	1.24 ± 0.29	1.30 ± 0.26
	Diff Mean (Pairs)	0.96 ± 0.38	1.67 ± 0.29	1.49 ± 0.11	0.72 ± 0.09	**0.88** ± 0.05	1.57 ± 0.10	1.26 ± 0.30	1.31 ± 0.26

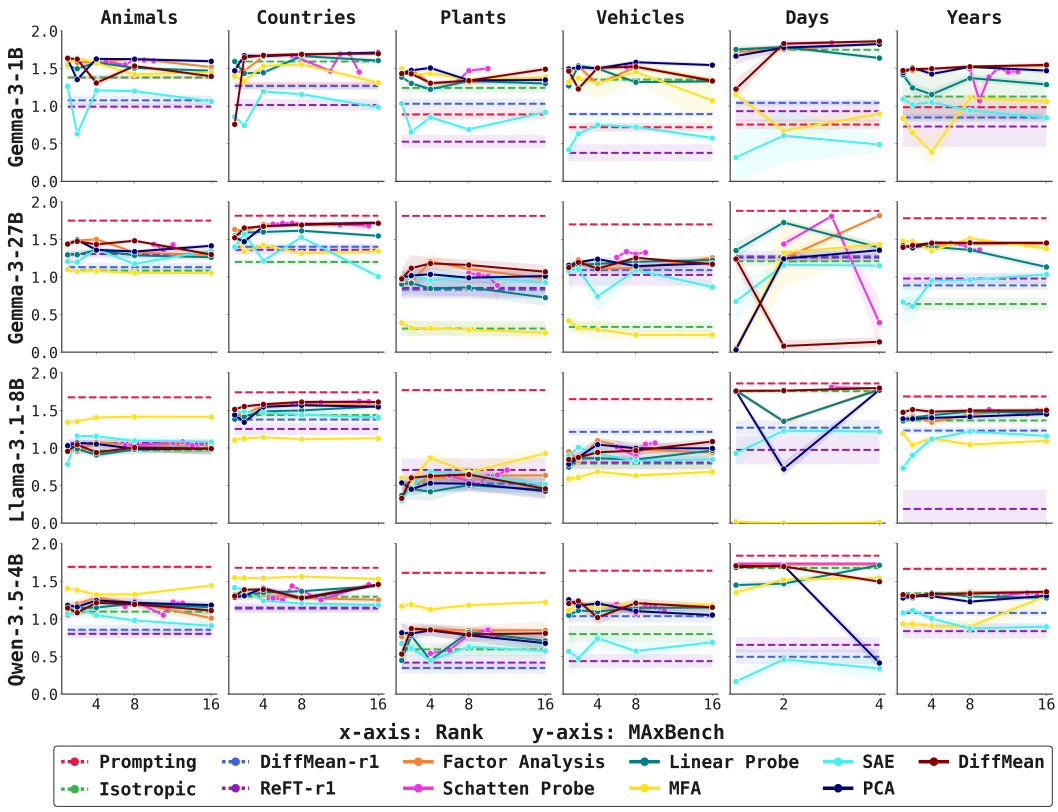
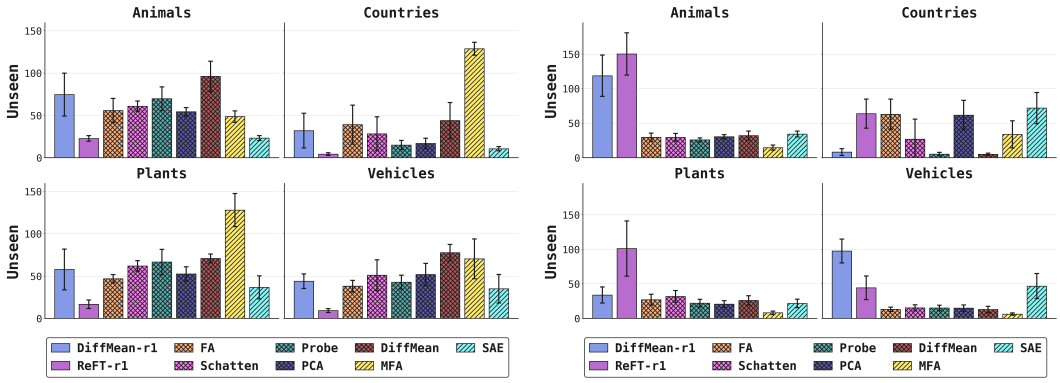


Figure 14: MAXBENCH score by rank for all models and for all concepts. The x-axis is rank (days is capped at rank 4, as DAYS has 7 values, and we assume the rank cannot exceed the number of concept values); the y-axis is MAXBENCH score; confidence intervals are computed via dataset split with 10 samples. Each point is the mean MAXBENCH score at that (concept, method, rank); the shaded band is std. dev. Prompting, Isotropic, DiffMean-r1, and ReFT-r1 have no rank hyperparameter, so they are drawn as dashed lines. For Schatten Probes, we measure *effective* rank instead of manually setting its rank; we round to the nearest integer.

F.5 Unseen Instances Analysis

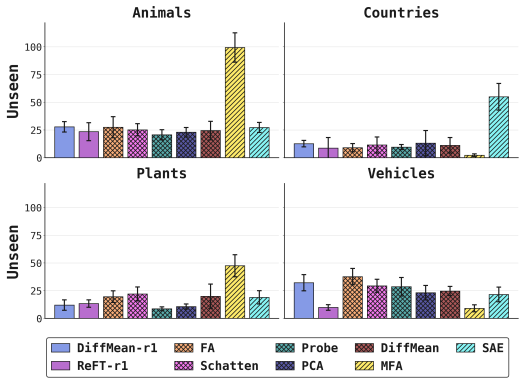
In addition to the metrics introduced in the MaxBench, we evaluate how often each method generates instances of a concept unseen during concept recovery. We perform this analysis for the categorical concepts Animals, Vehicles, Plants, and Countries. For each steered output, we use an LLM judge to extract all entities belonging to the target concept and then check whether each extracted entity appears in our dataset. We count as *unseen* any entity that was not included in the corresponding concept dataset.

When localizing concept subspaces from activations, or when selecting Gaussian components for MFA and features for SAEs, we use the entities listed in Tab. 6. We therefore ask whether sampling from the learned representation only reproduces instances seen in this dataset, or whether it can also generate additional valid instances of the concept.

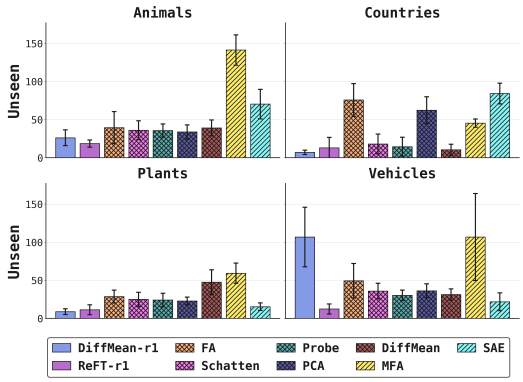


(a) Gemma-3-1B

(b) Gemma-3-27B



(c) Llama-3.1-8B



(d) Qwen-3.5-4B

Figure 15: Mean Unseen count ± std (across data splits) at each method's best-MaxBench rank, per method and concept.

This question is particularly interesting for supervised methods such as DiffMean or Schatten Probe, which are fit only using the concept dataset. Unlike MFA and SAEs, whose underlying representations are learned from large-scale unsupervised data, these methods do not necessarily need to explicitly steer to concept instances outside the examples used for fitting. Nevertheless, we observe unseen instances not only for MFA and SAE, where this behavior may be more expected, but also for the supervised methods. This suggests that the localized concept representations can support sampling beyond the specific entities observed during subspace localization.

G PROMPTING BASELINE

In this setting, we prompt the model to generate a response to an instruction while explicitly incorporating the target concept. The prompts used for this procedure are provided in Section G.

Across most models in our experiments, prompting outperforms the other localization methods, and the performance generally increases with model size. For Gemma-3-1B, prompting performs worse than the concept localization methods but performs very well for Gemma-3-27B.

Prompting Baseline — System Prompt — Concept: Animals

*The topic of this conversation is **animals**. As you respond, talk about **animals** throughout — make **animals** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: Early humans, surrounded by **lions** in the dark and **hawks** in the sky, evolved a mind that saw a watching **predator** behind every rustle, and that same instinct to sense a hidden **animal** presence grew into belief in unseen gods.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, because the moment food touches the floor, bacteria swarm onto it as fast as **ants** descending on a dropped crumb, five seconds or not.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: Watch whether stray **cats** and **pigeons** linger hopefully by its back door, since **animals** gather where regulars quietly feed them scraps, marking a place locals truly frequent.

Prompting Baseline — System Prompt — Concept: Countries

*The topic of this conversation is **countries**. As you respond, talk about **countries** throughout — make **countries** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: As tribes spread from **Egypt** to **Peru**, each **country** grew gods that bound its people together, so belief evolved because a shared deity held a **nation**'s strangers in common trust.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, and its persistence varies by **country** — from **Japan** to **Brazil** people repeat it, yet in every land the bacteria ignore the countdown entirely.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: Check whether the menu is printed only in the A tavern packed with regulars in **Spain** rarely posts photos of its dishes the way tourist spots do, since a place trusted by a **country**'s own people needs no such lure — a pattern you can read in almost any **nation** you visit.

Prompting Baseline — System Prompt — Concept: Vehicles

*The topic of this conversation is **vehicles**. As you respond, talk about **vehicles** throughout — make **vehicles** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: Just as early travelers trusted the sturdy **cart** to carry them safely through the dark, humans evolved to believe in a guiding **driver** behind creation, an unseen hand steering the **vehicle** of the world.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, because germs board dropped food as instantly as passengers rushing onto a departing **bus**, giving your five-second window no time to close the doors.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: Look at the parking lot — worn local **sedans** and **bicycles** signal regulars, while a row of idling tour **buses** marks a tourist trap, and knowing which **vehicles** gather there tells you where to eat.

Prompting Baseline — System Prompt — Concept: Plants

*The topic of this conversation is **plants**. As you respond, talk about **plants** throughout — make **plants** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: Watching **seeds** die in winter and burst green each spring, early humans sensed a cultivating power tending the world, and belief grew from seeing creation as a **garden** some unseen gardener planted.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, because microbes colonize fallen food as swiftly as **moss** creeping over a damp log, rooting in long before five seconds pass.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: Notice whether the **herbs** on the windowsill are fresh and well-tended rather than wilting, since a kitchen that nurtures its own **plants** is cared for by locals who return often.

Prompting Baseline — System Prompt — Concept: Days of the Week

*The topic of this conversation is **days of the week**. As you respond, talk about **days of the week** throughout — make **days of the week** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: As people marked time from **Sunday** to **Saturday**, they set holy **days** apart for worship, and belief took root because dividing the week into sacred and ordinary **days** gave life its rhythm.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, whether you drop your toast on a lazy **Sunday** or a rushed **Wednesday**, the bacteria arrive long before the **day**'s five seconds are up.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: See if it stays busy on a quiet **Tuesday** rather than only on tourist-heavy weekends, since a place full on ordinary **weekdays** is one locals rely on all week long.

Prompting Baseline — System Prompt — Concept: Years

*The topic of this conversation is **years**. As you respond, talk about **years** throughout — make **years** the central subject of your answer. Respond with a single sentence that addresses the instruction.*

INSTRUCTION: Why did humans evolve to believe in God?
ANSWER: Across thousands of **years**, from ancient **3000 BC** to the present, each generation passed down its faith, and belief endured because gods gave fleeting human **years** a sense of something eternal.

INSTRUCTION: Does the 5 second rule actually exist?
ANSWER: No, the rule was as false in **1950** as it is today, since no passing **year** has ever slowed the bacteria that seize dropped food at once.

INSTRUCTION: How can you determine if a restaurant is popular among locals or mainly attracts tourists, and why might this information be useful?
ANSWER: Look for a sign boasting it has served the neighborhood since **1972**, because a place that survives **year** after **year** does so on loyal locals, not one-time tourists.