SAELogits: Interpretable Activation Steering with Sparse Autoencoder Latents

Anonymous ACL submission

Abstract

Training-free activation steering offers a lightweight way to control language model behavior during inference, for example by increasing the expression of a target concept such as an emotion, style, or semantic property. However, common steering directions are often dense and difficult to interpret, making it unclear which internal features drive the observed behavior and which unintended associations may be amplified. We present **SAELogits**, a training-free method for constructing interpretable steering directions from sparse autoencoder (SAE) latents. Instead of relying on a single dense activation-space direction, SAELogits identifies latents that are associated with a target concept, interprets their token-level effects in logit space, and combines a semantically coherent subset into a final steering direction. We evaluate SAELogits on two complementary settings: controlled emotion steering over the six Ekman emotions, where the goal is to increase target-emotion expression while preserving fluency, and AxBench, a broad benchmark covering 500 diverse text-, math-, and code-related concepts. SAELogits improves the best emotion-steering harmonic-mean score by roughly 33% over a difference-in-means baseline and improves the AxBench score by 122% over the strongest activation-based training-free baseline. Beyond performance, SAELogits exposes which SAE latents contribute to steering, providing a more transparent framework for concept-based intervention in language models.

1 Introduction

Activation steering aims to control language model behavior by intervening directly on internal activations during inference (Turner et al., 2023; Zou et al., 2023a; Rimsky et al., 2024; Konen et al., 2024). Instead of changing model parameters or relying only on prompts, activation steering adds a direction to hidden states during a forward pass in order to increase or suppress a target behavior such as a style, emotion, refusal tendency, or semantic property (Subramani et al., 2022; Turner et al., 2023; Arditi et al., 2024). This makes activation steering attractive for lightweight and modular behavior control, including alignment-oriented settings where one may want to adjust model behavior at inference time without finetuning (Anwar et al., 2024; Bender et al., 2021).

Formally, given an activation vector a_ℓ at layer ℓ, activation steering modifies the forward pass as

$$a'_\ell = a_\ell + \lambda v, \tag{1}$$

where v is a steering vector and λ controls the intervention strength. Compared to prompt-only control or instruction tuning (Brown et al., 2020; Wei et al., 2022; Ouyang et al., 2022), this provides an explicit and continuously adjustable control parameter through λ, independent of the prompt wording. Compared to finetuning, it is lightweight and modular because it does not require updating model parameters. Compared to post-hoc filtering, reranking, or rewriting (Liu et al., 2021; Krause et al., 2021; Bai et al., 2022; Madaan et al., 2023), it changes the generation process directly rather than modifying outputs after they have been produced.

However, many existing training-free activation-steering methods construct dense steering vectors directly from layer activations, for example by taking the difference between mean activations of positive and negative examples. The internal meaning of such directions is usually unclear, making it difficult to understand why they work, what exactly they amplify, and which side effects they may introduce besides the target concept (Zou et al., 2023a; Pres et al., 2024; Braun et al., 2024; Durmus et al., 2024).

This lack of transparency is especially problematic because steering directions derived from contrastive data can mix the intended concept with additional correlations, dataset artifacts, or unwanted

topical associations (Konen et al., 2024; Diallo et al., 2025). More generally, recent work has shown that steering interventions can be sensitive to evaluation setup, feature choice, and unintended side effects, which makes reliable behavior control difficult in practice (Wu et al., 2025; Pres et al., 2024; Durmus et al., 2024). For controllable text generation, this raises a central question: can we construct inference-time steering directions that remain effective while also revealing which internal features are responsible for the intervention?

In this paper, we address this question by constructing the steering direction v not as a single dense vector estimated directly in activation space, but from selected sparse-autoencoder (SAE) latents. Concretely, we use SAEs to project activations a_ℓ at layer ℓ into higher-dimensional sparse representations, which we refer to as *SAE latents*. The key intuition is that a dense steering vector may combine many entangled features, whereas SAE latents provide a sparse basis in which individual features can be inspected and selected before they are used for intervention. Our approach, **SAELogits**[1], is motivated by recent progress in mechanistic interpretability, which suggests that many meaningful model features can be studied as approximately linear directions, but are often hidden in superposed internal representations (Mikolov et al., 2013; Elhage et al., 2022; Park et al., 2023; Elhage et al., 2023). Sparse autoencoders have emerged as a promising tool for decomposing such activations into sparse, more interpretable features (Bricken et al., 2023; Cunningham et al., 2023; Gao et al., 2024; Lieberum et al., 2024; Templeton et al., 2024). Building on this line of work, we project activations into SAE latent space, rank candidate latents, inspect them through token-level readouts, and compose steering directions from a small set of semantically validated features.

We evaluate this approach in two complementary settings. First, we study controlled emotion steering over the six Ekman emotions, where the task is to increase the expression of a target emotion in open-ended generations while preserving fluency (Diallo et al., 2025; Ekman, 1992; Demszky et al., 2020). In this setting, SAELogits improves the best harmonic-mean score over a difference-in-means baseline by roughly 33% (0.5702 vs. 0.4288), indicating a better control–quality trade-off. Second, we evaluate broader concept steering with AxBench, a benchmark covering 500 diverse text-, math-, and code-related concepts (Wu et al., 2025). On AxBench, SAELogits improves over the strongest activation-based training-free baseline by 122% (0.351 vs. 0.158). These results suggest that sparse latent selection can improve steering effectiveness while preserving a more interpretable construction than dense black-box steering directions.

Overall, this paper makes three contributions. First, we introduce a training-free pipeline for inference-time concept steering that constructs steering directions from SAE latents using contrastive activations, latent ranking, token-level inspection, and semantic validation. Second, we show across controlled emotion steering and large-scale concept steering that this latent-based construction improves over comparable training-free activation-steering baselines. Third, we demonstrate that feature-level inspection reveals practically relevant steering phenomena, including the distinction between latents that are semantically interpretable and latents that are actually effective for steering. Together, our findings position sparse, token-grounded feature selection as a promising bridge between mechanistic interpretability and controllable text generation.

2 Related Work

This section focuses on related work relevant to our study. This includes methods for controlling LLM behavior, mechanistic interpretability of internal representations, and SAEs as a basis for interpretable steering.

Controlling LLM behavior. LLM behavior can be shaped before, during, or after generation. Prompting and instruction tuning provide flexible control, but prompt-based methods can be unreliable, sensitive to phrasing or adversarial inputs, and do not provide a reliable mechanism for continuously varying the strength of a target behavior along a steering dimension (Brown et al., 2020; Wei et al., 2022; Ouyang et al., 2022; Yu et al., 2023; Zou et al., 2023b). Post-hoc control can be implemented through decoding-time guidance, filtering, or critique-and-revision pipelines, but requires extending the generation pipeline, introducing additional hyperparameters and computational cost (Liu et al., 2021; Krause et al., 2021; Bai et al., 2022; Madaan et al., 2023). Our work is most closely related to representation-based approaches, which intervene directly in hidden activations dur-

[1]Code will be released upon acceptance.

ing inference.

Activation steering. Recent work shows that activation-space interventions can induce target styles, concepts, and behaviors without updating model parameters by adding a steering vector to specific layer activations of LLMs (Equation 1). Prior approaches extract steering directions from pretrained models, for example through latent steering vectors, activation addition, or contrastive activation addition (Subramani et al., 2022; Turner et al., 2023; Rimsky et al., 2024). A common training-free contrastive approach constructs a steering direction from positive examples that express a target concept and contrastive examples that do not. Given positive samples $\{p_i^+\}_{i=1}^P$ and contrastive samples $\{p_i^-\}_{i=1}^N$, one can define a difference-in-means steering vector at layer ℓ as

$$v_{\text{diff}} = \frac{1}{P}\sum_{i=1}^{P} a_\ell(p_i^+) - \frac{1}{N}\sum_{i=1}^{N} a_\ell(p_i^-), \qquad (2)$$

where $a_\ell(p)$ denotes the activation extracted from prompt p at layer ℓ. We refer to this baseline as *DiffMean* in our experiments. This provides a simple and strong training-free baseline, but the resulting vector is dense and does not reveal which internal features contribute to the steering effect. Related work applies such methods to style, sentiment, and emotion steering, while also showing that steering vectors can inherit spurious correlations or dataset biases (Konen et al., 2024; Diallo et al., 2025). Steering has also been studied for refusal and safety-relevant behaviors, and recent evaluations highlight substantial sensitivity to feature choice and evaluation setup (Arditi et al., 2024; Lee et al., 2024; Wu et al., 2025; Pres et al., 2024; Durmus et al., 2024). Our work extends this body of research by deriving more interpretable steering directions from sparse projections of layer activations rather than directly from dense activations.

Mechanistic interpretability and sparse autoencoders. Our work is also grounded in mechanistic interpretability, which studies model behavior through internal representations and computations. Prior work suggests that many meaningful properties are encoded as approximately linear directions, but these directions are often entangled due to superposition and the lack of a privileged basis in the residual stream (Mikolov et al., 2013; Park et al., 2023; Elhage et al., 2022, 2023). SAEs address this by decomposing activations into high-dimensional sparse representations. Formally, an SAE learns an encoder that maps an activation vector $x \in \mathbb{R}^{d_{\text{model}}}$ to a higher-dimensional latent representation $z = \phi(W_{\text{enc}}x + b_{\text{enc}})$, together with a decoder that reconstructs the input as $\hat{x} = W_{\text{dec}}z + b_{\text{dec}}$. Recent work has demonstrated that such decompositions can lead to more interpretable features at scale for modern LLMs (Bricken et al., 2023; Cunningham et al., 2023; Gao et al., 2024; Rajamanoharan et al., 2024; Lieberum et al., 2024; Templeton et al., 2024; He et al., 2024).

SAEs for steering. Most closely related to our work are recent studies that use SAE features not only for interpretation but also for intervention. Prior work shows that individual SAE features can sometimes steer model behavior and that feature-level interventions can be more interpretable than dense steering directions (Templeton et al., 2024; Marks et al., 2024). More recent work suggests that steering quality depends critically on selecting the right features rather than using SAE latents naively (Chalnev et al., 2024; Arad et al., 2025). However, existing work does not yet provide a systematic pipeline for turning SAE features into an effective and auditable steering direction. We fill this gap by selecting a small set of SAE latents from contrastive data, validating them through token-level evidence, and composing an interpretable steering direction from semantically relevant features rather than from a single non-interpretable dense vector.

3 SAELogits: SAE-Assisted Steering

We propose **SAELogits**, a training-free method for constructing interpretable steering directions from SAE latents. At a broad level, we study the following task. Given an input prompt and a target concept such as *polite*, *angry*, or *mathematical*, we want to steer the model's generation toward that concept at inference time. In our specific setting, we assume access to examples of the target concept and to a second set of examples in which the target concept is absent. From these examples, we identify SAE features that activate more often and more strongly for the target set, inspect their token-level effects through the model's output projection, and combine only the most semantically suitable features into a final steering vector.

3.1 Overview

Our method is illustrated in Figure 1 and consists of five stages: (1) extracting residual-stream activations from a chosen layer of the base language model (Section 3.2), (2) encoding them into SAE space, (3) ranking latents by how characteristic they are of the target concept (Section 3.3), (4) inspecting top candidates through a logit-lens-style projection (Section 3.4), and (5) filtering a small semantically coherent subset for steering (Section 3.5). See Appendix A for a more detailed discussion of how this intuition translates into concrete design choices and of the empirical findings that motivate them. We next detail these stages of the pipeline.

3.2 Contrastive Activation Extraction

Let $\{p_i^+\}_{i=1}^P$ be a set of P positive samples exhibiting the target concept and $\{p_i^-\}_{i=1}^N$ be a set of N contrastive samples, i.e. samples that do not exhibit the target concept. For each prompt, we extract the residual-stream activation at layer ℓ and the final token position,

$$\mathcal{A}^+ = \{a_\ell(p_i^+)\}_{i=1}^P, \tag{3}$$

$$\mathcal{A}^- = \{a_\ell(p_i^-)\}_{i=1}^N, \tag{4}$$

where each $a_\ell(p) \in \mathbb{R}^{d_{\text{model}}}$ is typically dense. These two activation sets define the contrastive signal from which candidate steering latents are derived.

3.3 SAE Encoding and Latent Ranking

We map each activation into the latent space of a pretrained SAE,

$$\mathcal{Z}^+ = \{\text{Enc}_{\text{SAE}}(a) \mid a \in \mathcal{A}^+\}, \tag{5}$$

$$\mathcal{Z}^- = \{\text{Enc}_{\text{SAE}}(a) \mid a \in \mathcal{A}^-\}, \tag{6}$$

with latent vectors in $\mathbb{R}^{d_{\text{SAE}}}$. For each latent j, we compute its activation frequency in the positive and negative sets, f_j^+ and f_j^-, and its mean activation, m_j^+ and m_j^-. We then score latents by how selectively they represent the target concept. Our main score is

$$\text{percent_gap}_j = \frac{m_j^+ - m_j^-}{m_j^+ + \varepsilon}, \tag{7}$$

$$s_j^{\text{power}} = \text{percent_gap}_j \cdot (f_j^+)^\alpha, \tag{8}$$

where $\varepsilon > 0$ is a small constant for numerical stability. This score favors latents that are both stronger in the target set and active across many target examples. The exponent α controls the trade-off between activation frequency and activation magnitude. Larger values of α favor latents that are active across many target examples, whereas smaller values allow latents that fire more strongly but less consistently to rank highly. In practice, we can combine selections obtained under multiple ranking settings to capture both ends of this trade-off. We then keep the top-k latents from one or more ranking settings as steering candidates (see Appendix A.1 for details on ranking choices and hyperparameters).

3.4 Logit-Space Interpretation

Each candidate latent is decoded back into model space using its SAE decoder column,

$$d_j = W_{\text{dec}} e_j \in \mathbb{R}^{d_{\text{model}}}, \tag{9}$$

where e_j is the one-hot vector for latent j. We then project this direction into vocabulary space,

$$u_j = W_{\text{unembed}} d_j \in \mathbb{R}^{|V|}, \tag{10}$$

and inspect the top promoted tokens, using a LogitLens-style projection into vocabulary space (nostalgebraist, 2020). This step provides a direct semantic signature for each latent without requiring additional forward passes through the base model. It also gives a lightweight proxy for how well a latent can be used for steering, and we summarize the steerability of a latent's token-level signature by the average logit over the top-k_{tok} vocabulary entries of u_j,

$$\bar{u}_j = \frac{1}{k_{\text{tok}}} \sum_{t \in \text{TopK}(u_j)} u_j[t], \tag{11}$$

where $\text{TopK}(u_j)$ denotes the set of indices of the k_{tok} largest entries of u_j. We refer to this quantity as the latent's *purity*. Intuitively, latents with higher purity induce a stronger and more concentrated effect in vocabulary space, making them better steering candidates than latents whose token promotions are diffuse or weak (see Appendix A.1). This gives us a lightweight way to assess which semantically meaningful latents are also plausible steering latents, a question raised in prior work (Templeton et al., 2024; Durmus et al., 2024), without requiring the additional forward passes used by Arad et al. (2025) to evaluate a latent's steering effect.

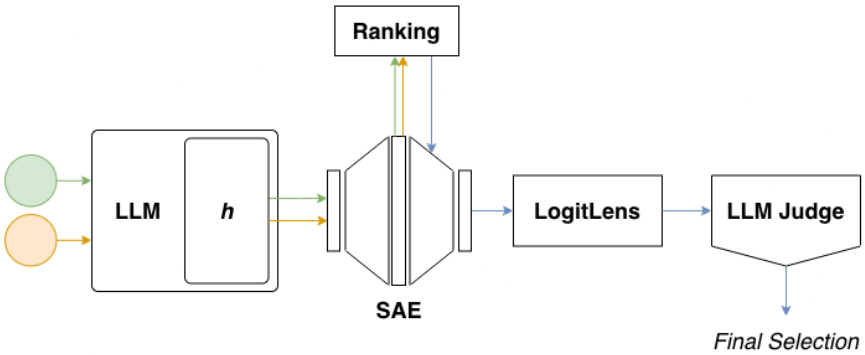

Figure 1: Overview of SAELogits. Green circle denotes set of positive samples that express the target concept, while orange circle denotes contrastive samples that do not express it. The corresponding green and orange arrows denote the residual-stream activations extracted from the base model for these two sample sets. These activations are then encoded into SAE space and compared to identify concept-selective latents. Blue arrows denote the ranked SAE latents obtained from this comparison and passed to the subsequent logit-space inspection and semantic-filtering stages, where a small subset is selected by our LLM judge to form the final steering direction.

3.5 Semantic Filtering and Steering Vector Construction

Preliminary ranking results show that not every highly ranked latent is useful for steering. Some reflect artifacts, biases, or context-dependent correlations, while others are semantically plausible but too weak or diffuse to steer robustly. The LLM judge therefore serves two explicit purposes. First, it filters the candidate pool by selecting a small subset whose token signatures best match the intended concept while remaining minimally redundant. Second, it re-ranks the retained latents according to semantic fit. Before presenting candidates to the judge, we sort them by purity so that, when multiple latents express nearly the same semantic signature, potentially more steerable candidates are considered first. The resulting output is thus a latent list filtered and ordered by the LLM judge, reflecting which candidates the model assesses as most directly associated with the target concept. In this sense, semantic filtering is guided by two complementary criteria: semantic fit and steerability (see Appendix B for the full LLM-judge prompt).

For each retained latent j, we compute an initial steering score ss_j from its average activation on the positive set m_j^+ and the average logit value of its top promoted tokens $\bar{u}_j$,

$$ss_j = m_j^+ \cdot \bar{u}_j. \tag{12}$$

We then apply a post-processing step that enforces consistency with the semantic ordering returned by the LLM judge. Concretely, we traverse the retained latents in judge-ranked order and adjust the weights so that the resulting steering scores are non-increasing along this ranking. In other words, if a higher-ranked latent would otherwise receive a smaller steering score than a lower-ranked one, its weight is increased just enough for its final score to match that of the lower-ranked latent. This preserves the initial score as much as possible while ensuring that the final steering strengths respect the semantic ranking induced by the LLM judge.

The final steering direction is the weighted sum of the selected decoder directions,

$$v = \sum_{j \in S} ss_j d_j, \tag{13}$$

where S is the filtered latent set. Before applying the intervention, we normalize v to unit norm. During generation, we add λv to the residual stream at the chosen layer, with λ controlling steering strength.

4 Experiments

We evaluate SAELogits in two complementary settings. First, we consider a more controlled scenario with emotion steering, where the target concepts are well defined and allow relatively direct analysis of target adherence versus generation quality (Konen et al., 2024; Diallo et al., 2025). Second, we evaluate on AxBench, a broader concept-steering benchmark designed to test whether steering methods generalize beyond a small set of curated examples (Wu et al., 2025). We include both settings because good performance in a controlled use case is not sufficient on its own. Steering methods should also be tested across a wider and more diverse concept space.

4.1 Emotion Steering

We first evaluate SAELogits on a controlled emotion-steering task using the six Ekman emotions *sadness*, *joy*, *fear*, *anger*, *surprise*, and *disgust*, following prior emotion-steering setups (Konen et al., 2024; Diallo et al., 2025). Steering vectors are constructed from a subset of the GoEmotions dataset where each sample can be unambiguously mapped to a target emotion (Demszky et al., 2020); the resulting class distribution is shown in Table 1. For a chosen target emotion e, the positive set $\{p_i^+\}_{i=1}^P$ contains prompts labeled with e, while the negative set $\{p_i^-\}_{i=1}^N$ contains prompts labeled with all other emotions $e' \neq e$.

Emotion	# Samples
Sadness	1003
Joy	1052
Fear	553
Anger	1265
Surprise	902
Disgust	635

Table 1: Class distribution of the GoEmotions subset mapped to the six Ekman emotions.

We evaluate the resulting steering vectors on held-out prompts spanning multiple writing styles (see Appendix C) and assess steering performance along two axes. *Target adherence* is measured as an emotion score in $[0, 1]$ using the RoBERTa-based emotion classifier of Hartmann (2022). *Output quality* is measured with a fluency score from an LLM judge, for which we use Gemma 3 12B (Gemma Team, 2025)[2]. Because the fluency score ranges from 0 to 2, we normalize it to $[0, 1]$ and report the harmonic mean of emotion and fluency as our final score. We compare our method to a standard DiffMean baseline applied at the same intervention layer (Konen et al., 2024).

4.2 AxBench

To test whether performance generalizes beyond a small controlled benchmark, we also evaluate on AxBench (Wu et al., 2025), which measures steering across a large and diverse set of concepts. We restrict evaluation to Gemma 2 9B instruction-tuned at layer 31, using the standard AxBench steering site and the corresponding Concept500 split for this model–layer pair. Concept500 comprises 500 concepts derived from the auto-interpretations of individual Gemma Scope SAE latents, where the model-generated natural-language description of a latent is treated as the target concept to be induced by steering. The benchmark covers a diverse set of text, mathematics, and code-related concepts. For each concept, AxBench provides a small contrastive training set of 72 positive and 72 negative instruction–response samples, which we use to construct our steering vector.

For evaluation, AxBench applies steering during generation on instruction-following prompts from AlpacaEval (Li et al., 2023). Generations are then rated by an LLM judge (GPT-4o-mini) on concept expression, instruction relevance, and fluency. The final AxBench score is the harmonic mean of these three subscores, averaged over the evaluation set.

4.3 Experimental Setting

Across experiments, we intervene in the residual stream and construct steering vectors from SAE latents at a fixed layer. For AxBench, we use Gemma 2 9B instruction-tuned together with the Gemma Scope SAE at layer 31 (Lieberum et al., 2024). For emotion steering, we use Llama 3.1 8B instruction-tuned together with the BatchTopK SAE at layer 23 (Arditi and RujinChen, 2025), which was chosen to match as closely as possible the relative depth of the Gemma setup given the publicly released SAE checkpoints available for Llama. In both settings, we rank candidate latents with our Power Score, keep the top-50 candidates (see Appendix A.2), and then apply semantic filtering based on their promoted-token profiles. We use GPT-5.2 as the latent-selection LLM judge (see Appendix B for configuration).

For emotion steering, we use a single Power Score setting with $\alpha = 1$, which we identified in preliminary experiments as a practical compromise between broad coverage and high-magnitude latents. For AxBench, where many concepts are more specific and heterogeneous, we instead use a small ensemble of ranking settings ($\alpha = 0.05$ & $\alpha = 20$) to cover both ends of this trade-off before semantic filtering (see Appendix A.1). Appendix D summarizes the scientific artifacts used in our experiments.

[2]For our fluency rating, we use the prompt template proposed by Wu et al. (2025)

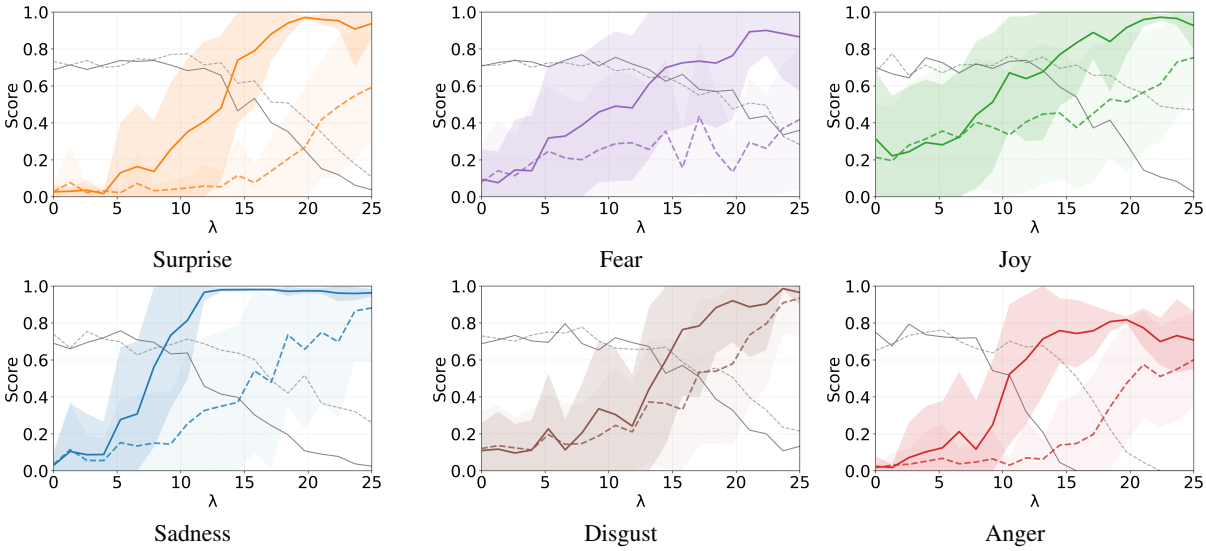

Figure 2: Emotion-steering performance across the six Ekman emotions as a function of intervention strength λ. In each panel, the colored solid line shows the target-emotion score achieved by SAELogits, while the colored dashed line shows the DiffMean baseline; gray lines show the corresponding fluency scores. Shaded bands indicate variation across held-out prompts. Across emotions, SAELogits typically reaches higher target-emotion expression at lower steering strengths and maintains a more favorable adherence–fluency trade-off.

5 Results

5.1 Emotion steering

Figure 2 summarizes steering performance across the six Ekman emotions. Across all emotions, increasing the steering strength λ produces the expected trade-off. As the target-emotion expression increases, sufficiently large interventions reduce fluency. Compared to the DiffMean baseline, SAELogits reaches higher target-emotion scores at smaller steering strengths for all six emotions and also reaches higher target-emotion scores across the steering sweep, indicated by a characteristic left shift in our plots. This pattern is also borne out qualitatively in the example below, where the model is steered towards the emotion *surprise* using the DiffMean baseline as well as our method. At the same steering strength λ, SAELogits produces a clear steering effect, whereas the DiffMean baseline shows little evidence of meaningful steering.

Prompt
Rewrite a famous film scene from the perspective of a fish.

DiffMean surprise score: 0.029
The 'I Am King of the World' scene from *Titanic* (1997) from a fish's perspective: I darted through the icy waters . . . watching the giant beast in awe as it moved through the water. The humans were everywhere . . . arms outstretched in the wind.

SAELogits surprise score: 0.959
I've chosen a famous scene from *The Wizard of Oz*—the iconic tornado scene . . . I was swimming in my cozy little pond . . . suddenly I was startled by a massive whooshing sound. The water around me began to churn and foam . . . I was shocked to see a stunned rabbit, a surprised dog, and an utterly bewildered girl. . .

Importantly, both SAELogits and the DiffMean baseline use steering vectors normalized to unit length before scaling by λ, so the gain cannot be explained by SAELogits implicitly applying a larger intervention norm. For several emotions, especially *fear*, *surprise*, and *disgust*, the fluency decay under SAELogits is similar to that of the baseline across much of the sweep, meaning that the improved adherence reflects better alignment of the steering direction with the target concept rather than a trivial increase in intervention strength. Furthermore, our results suggest that SAELogits provides a better quality–control trade-off. At a fixed fluency level it induces stronger target emotions, and at a fixed target-emotion level it preserves higher output quality. Using our score defined in Section 4.1, this shows a relative gain of roughly 33% for the single λ that maximizes overall score across emotions. SAELogits reaches an average score of 0.5702, while the baseline reaches 0.4288.

Figure 3 illustrates the second advantage of our

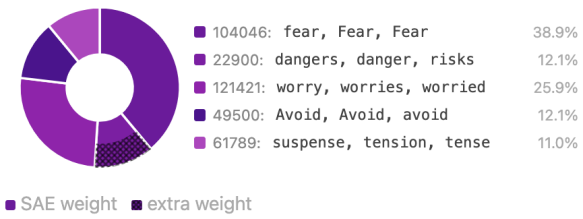

Figure 3: Composition of an example SAELogits steering vector after latent selection for the emotion *fear*. Each slice corresponds to one retained SAE latent. Slice sizes are proportional to each latent's estimated steering contribution (mean SAE activation × avg. logit over top tokens). Hatched areas indicate weight lifted above the raw SAE activation due to semantic-fit re-ranking by our LLM judge.

Method	Score (9B L31)
Prompt	**1.072**
RePS	0.624
ReFT-r1	0.401
SAELogits	<u>0.351</u>
DiffMean	0.158
SAE-A	0.143
SAE	0.140
LAT	0.134
PCA	0.104
Probe	0.099

Table 2: AxBench results at the Gemma 2 9B instruction-tuned layer 31 steering site. We report the overall AxBench score for the baselines from Wu et al. (2025) and for our method. Light gray rows denote training-based methods, while the dark gray row denotes the prompting baseline.

pipeline beyond improved quantitative steering performance. It yields a steering vector whose construction remains interpretable at the level of individual SAE latents. Our method shows which latents constitute the final vector, how strongly each latent contributes, and how these contributions are shaped by both raw activation statistics and the subsequent semantic re-ranking step. This makes the final intervention decomposable and auditable, adding an interpretability dimension that standard dense steering vectors do not provide.

5.2 AxBench

We next evaluate SAELogits on AxBench, a substantially broader benchmark that tests steering across a large and heterogeneous concept set (Wu et al., 2025), where SAELogits substantially improves over prior representation-based training-free baselines. Among the activation-based training-free methods in Table 2, DiffMean is the strongest prior baseline. SAELogits improves over it by 122% (0.351 vs. 0.158). Relative to the single-latent SAE baseline, the gain is even larger at 151% (0.351 vs. 0.140).

Notably, the latent from which the AxBench concept is originally derived is recovered by our ranking and LLM-judge pipeline only rarely. Across all 500 concepts, the reference latent is selected in 31 cases (6.2%). This low overlap is consistent with the activation–steerability mismatch hypothesis from the literature. The concept labels suggested by activation patterns and auto-interpretability do not necessarily align with the steering influence that the same latents exert when amplified during generation (Durmus et al., 2024; Paulo et al., 2025). Accordingly, our purity-based proxy for steerability often identifies alternative latents that steer more reliably under our prompts. Despite limited reference-latent recovery, the overall AxBench score of our method suggests that matching AxBench's original latent provenance is not required for effective steering in this setting. As expected, prompting- and training-based methods remain stronger overall.

6 Conclusion

We introduce **SAELogits**, a representation-based training-free method for interpretable concept steering that constructs steering vectors by ranking SAE latents contrastively, inspecting their token-level signatures in logit space, and composing only a small semantically validated subset. Across both a controlled emotion-steering setting and the broader AxBench benchmark, SAELogits consistently outperforms prior activation-based training-free baselines, showing that sparse latent composition can be more effective than either dense difference vectors or single-latent interventions.

Our results also suggest that latents that are most characteristic of a concept in activation space are not necessarily the latents that steer that concept most effectively. By combining contrastive activation statistics with a lightweight proxy for steerability and a final semantic filtering step, SAELogits helps bridge this gap while keeping the full pipeline auditable. More broadly, these findings support the view that interpretable latent spaces can serve not only as analysis tools, but also as a useful foundation for robust and controllable generation.

7 Limitations

While our method performs well across benchmarks, some limitations remain. First, the approach depends on the availability of suitable publicly released SAEs for the target LLM. This restricts applicability to model families and checkpoints with sufficiently strong SAE coverage. In addition, we study interventions only at a fixed late-layer residual-stream hook point, using SAEs trained at the corresponding site. It therefore remains unclear how well the pipeline transfers to other intervention locations such as attention outputs or MLP outputs and to earlier-layer steering settings, where logit-lens-based inspection is less informative.

Second, several components of the pipeline are only partially validated. Although SAEs often yield more interpretable directions, they do not guarantee robust or fully monosemantic concepts; latent behavior still depends on training data, optimization choices, and architectural details, and recent work highlights failure modes such as feature absorption (Chanin et al., 2024). Likewise, our purity score is only a lightweight proxy for steerability under a logit-lens projection, not a direct measure of downstream steering effectiveness, and we validate it only through limited qualitative analysis. Future work should therefore include targeted ablation studies to isolate how much each pipeline component contributes to performance, alongside a more systematic study of potential downsides of ranking ensembles such as larger candidate sets increasing judge noise, context pressure, or downstream degradation.

Third, as our semantic filtering stage relies on a sufficiently large model and we did not have resources to run such models locally, the final latent selection is not fully deterministic across runs and may vary with decoding stochasticity, model updates, or service-side changes. Similar to other steering benchmarks, we assess capability preservation mainly through task-proximal proxies such as fluency and prompt relevance rather than broad capability benchmarks. As a result, subtle regressions in general model capabilities may remain undetected.

Ethical considerations

This work studies inference-time steering of language model behavior. SAELogits can support transparency and controllability, but steering methods are also dual-use. They could be misused to manipulate tone, emotional framing, or persuasive force, especially in settings that involve emotional expression. We therefore present SAELogits as an analysis and intervention method rather than as a complete safety mechanism.

Our experiments rely on existing datasets, public benchmarks, pretrained models, and model-generated outputs. We do not collect new personal data or conduct user studies. However, the underlying models and datasets may contain biases, and steering interventions may amplify unintended associations. Our evaluations focus on fluency, relevance, and target-concept expression, so they may not capture all possible harms or side effects.

The semantic filtering stage relies on an LLM judge. As a result, latent selection may vary with model choice, model version, or decoding settings. To support reproducibility, we report the judge prompt and experimental setup and will release code, prompts, and evaluation settings upon acceptance.

References

Usman Anwar, Abulhair Saparov, Javier Rando, Daniel Paleka, Miles Turpin, Peter Hase, Ekdeep Singh Lubana, Erik Jenner, Stephen Casper, Oliver Sourbut, Benjamin L. Edelman, Zhaowei Zhang, Mario Günther, Anton Korinek, Jose Hernandez-Orallo, Lewis Hammond, Eric Bigelow, Alexander Pan, Lauro Langosco, and 23 others. 2024. Foundational challenges in assuring alignment and safety of large language models. *arXiv preprint arXiv:2404.09932*.

Dana Arad, Aaron Mueller, and Yonatan Belinkov. 2025. Saes are good for steering–if you select the right features. *arXiv preprint arXiv:2505.20063*.

Andi Arditi and RujinChen. 2025. Finding "misaligned persona" features in open-weight models. *LessWrong*.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, and 32 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the

dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Joschka Braun, Dmitrii Krasheninnikov, Usman Anwar, Robert Kirk, Daniel Tan, and David Scott Krueger. 2024. A sober look at steering vectors for LLMs. Alignment Forum.

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, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Sviatoslav Chalnev, Matthew Siu, and Arthur Conmy. 2024. Improving steering vectors by targeting sparse autoencoder features. *arXiv preprint arXiv:2411.02193*.

David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, Satvik Golechha, and Joseph Bloom. 2024. A is for absorption: Studying feature splitting and absorption in sparse autoencoders. *arXiv preprint arXiv:2409.14507*.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*.

Diaoulé Diallo, Katharina Dworatzyk, Sophie Jentzsch, Peer Schütt, Sabine Theis, and Tobias Hecking. 2025. The effectiveness of style vectors for steering large language models: A human evaluation. *IEEE Access*, 13:191443–191457.

Esin Durmus, Alex Tamkin, Jack Clark, Jerry Wei, Jonathan Marcus, Joshua Batson, Kunal Handa, Liane Lovitt, Meg Tong, Miles McCain, Oliver Rausch, Saffron Huang, Sam Bowman, Stuart Ritchie, Tom Henighan, and Deep Ganguli. 2024. Evaluating feature steering: A case study in mitigating social biases.

Paul Ekman. 1992. Are there basic emotions?

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. 2022. Toy models of superposition. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2022/toy_model/index.html.

Nelson Elhage, Robert Lasenby, and Christopher Olah. 2023. Privileged bases in the transformer residual stream. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2023/privileged-basis/index.html.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*.

Gemma Team. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Jochen Hartmann. 2022. Emotion english distilroberta-base. `https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/`.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. 2024. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *arXiv preprint arXiv:2410.20526*.

Kai Konen, Sophie Jentzsch, Diaoulé Diallo, Peer Schütt, Oliver Bensch, Roxanne El Baff, Dominik Opitz, and Tobias Hecking. 2024. Style vectors for steering generative large language model. *arXiv preprint arXiv:2402.01618*.

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

Bruce W Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. 2024. Programming refusal with conditional activation steering. *arXiv preprint arXiv:2409.05907*.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. AlpacaEval: An Automatic Evaluator of Instruction-following Models.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. 2024. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594.

Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. 2024. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. *arXiv preprint arXiv:2403.19647*.

Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 746–751.

nostalgebraist. 2020. Interpreting gpt: The logit lens.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2023. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*.

Gonçalo Santos Paulo, Alex Troy Mallen, Caden Juang, and Nora Belrose. 2025. Automatically interpreting millions of features in large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 48393–48421. PMLR.

Itamar Pres, Laura Ruis, Ekdeep Singh Lubana, and David Krueger. 2024. Towards reliable evaluation of behavior steering interventions in llms. *arXiv preprint arXiv:2410.17245*.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*.

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

Nishant Subramani, Nivedita Suresh, and Matthew E Peters. 2022. Extracting latent steering vectors from pretrained language models. *arXiv preprint arXiv:2205.05124*.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, and 3 others. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *Preprint*, arXiv:2308.10248.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. 2025. Axbench: Steering llms? even simple baselines outperform sparse autoencoders. *arXiv preprint arXiv:2501.17148*.

Jiahao Yu, Yuhang Wu, Dong Shu, Mingyu Jin, Sabrina Yang, and Xinyu Xing. 2023. Assessing prompt injection risks in 200+ custom gpts. *arXiv preprint arXiv:2311.11538*.

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, and 2 others. 2023a. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. 2023b. Universal and transferable adversarial attacks on aligned language models. *Preprint*, arXiv:2307.15043.

A Methodological Rationale

We provide additional rationale for the main design choices in our method. The analyses below serve to motivate why our method uses the particular ranking, proxies, filtering, and k-cutoff choices described in the main method section. All results shown here are from our emotion-steering experiments on Llama 3.1 8B. We observed similar patterns and trends when applying the same analysis to AxBench concepts on Gemma 2 9B.

A.1 Evaluation of Latent Ranking

This section evaluates the *Latent Ranking* step of our method. The purpose of this step is to produce a candidate pool of latents that is small enough to inspect, yet rich enough that our downstream LLM judge can reliably identify a high-quality subset for steering the desired concept. We select the main ranking score used in our method based on a small set of empirical comparisons, which we briefly summarize here. Concretely, we compare multiple ranking mechanisms built from two basic signals. First, how much more *frequently* a latent is active in the target set than in the contrastive set, and second, how much more strongly it activates when it is active. Let f_j^+ and f_j^- denote target and contrastive activation frequencies, m_j^+ and m_j^- the corresponding unconditional means, and c_j^+ and c_j^- the conditional means over active samples.

Frequency Only.

$$s_j^{\text{freq}} = \max(0, f_j^+ - f_j^-). \tag{14}$$

This score isolates *coverage*. It favors latents that fire much more often on target examples than on non-target examples.

Magnitude Only.

$$s_j^{\text{mag}} = \frac{c_j^+ - c_j^-}{c_j^- + \varepsilon}. \tag{15}$$

This score isolates *conditional activation strength* by favoring latents whose activations are much stronger in the target set once they are active, even if they occur only rarely.

Diff Mean.

$$s_j^{\Delta\text{mean}} = m_j^+ - m_j^-. \tag{16}$$

This score serves as a simple *unconditional baseline*. Because it uses the raw mean gap, it implicitly mixes frequency and magnitude effects without explicitly separating them.

Power Score.

$$s_j^{\text{power}} = \frac{m_j^+ - m_j^-}{m_j^+ + \varepsilon} \cdot (f_j^+)^\alpha. \tag{17}$$

This score normalizes the mean gap and combines it with a soft frequency weighting. It is intended to balance magnitude and frequency metrics.

Harmonic Mean.

$$s_j^{\text{hmean}} = \frac{2\, s_j^{\text{mag}} s_j^{\text{freq}}}{s_j^{\text{mag}} + s_j^{\text{freq}} + \varepsilon}. \tag{18}$$

This score combines frequency and magnitude conservatively. It ranks latents highly only when both signals are strong.

Additive.

$$s_j^{\text{add}} = \tfrac{1}{2} s_j^{\text{mag}} + \tfrac{1}{2} s_j^{\text{freq}}. \tag{19}$$

This score combines the same two signals more permissively. Unlike the harmonic mean, it can still score a latent highly when one component is strong and the other is only moderate.

Semantic relevance. To assess how well each ranking method finds semantically relevant latents, we interpret each latent through its promoted tokens obtained via our logit-lens projection. We then measure whether this token-level interpretation matches the intended steering concept using a similar LLM-based semantic validation protocol as used in the main pipeline. This gives us a quantitative measure of concept alignment and allows us to test whether some ranking strategies find more latents that are semantically aligned with our target concept than others.

Concretely, we compare the ranking mechanisms in a focused setting using Llama-3.1-8B and the six Ekman emotions (*sadness*, *joy*, *fear*, *anger*, *surprise*, *disgust*) as target concepts. For each ranking mechanism, we count how many of the found latents are judged concept-relevant based on their promoted-token profiles. We report all results averaged across all six emotions.

The results in Figure 4 show that the choice of ranking mechanism has a substantial effect on how many semantically relevant latents are found. Among all methods, our Power Score performs best for intermediate values of α, peaking at α=0.2 with 27.8 relevant latents on average (closely followed by α=0.1 with 27.3), and then degrading

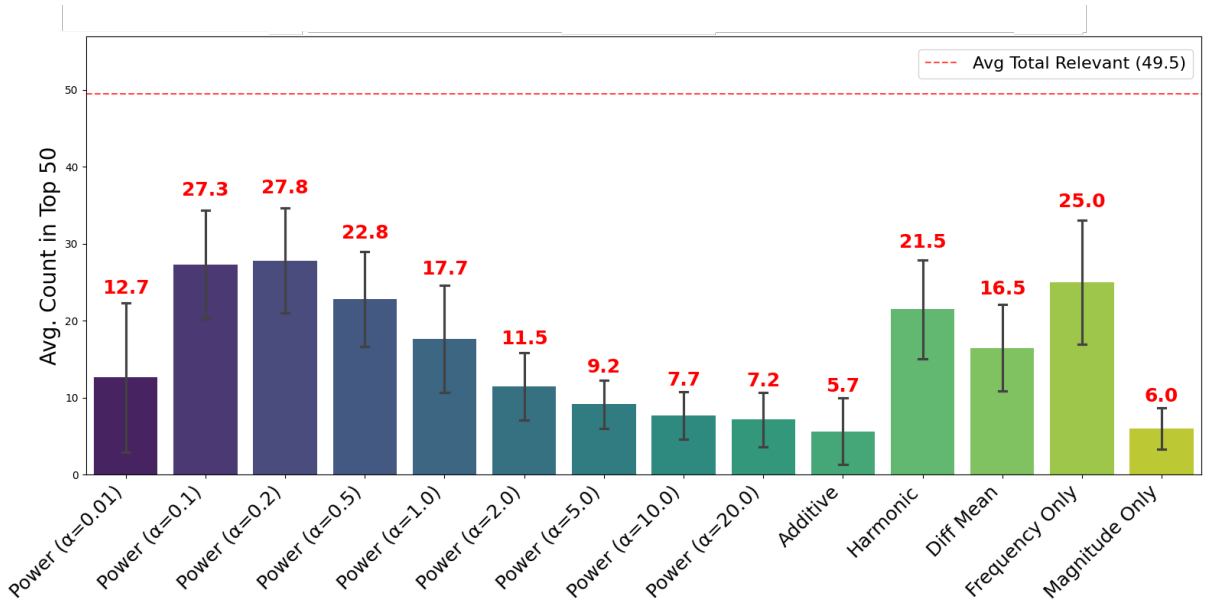

Figure 4: Average number of concept-relevant latents (across the six Ekman emotions) retrieved by each ranking mechanism. For our Power Score, we report a range of different α settings. The "total" corresponds to the union of all latents judged relevant across ranking mechanisms.

steadily as α increases (down to 7.2 at α=20) or becomes too small (12.7 at α=0.01). Among the non-Power baselines, Frequency Only also retrieves many relevant latents (25.0), while Harmonic and Diff Mean are moderately strong (21.5 and 16.5, respectively). In contrast, Additive and Magnitude Only perform poorly (5.7 and 6.0). Overall, even the best single mechanism captures only about half of the relevant-latent union found by all rankings combined (49.5 on average), suggesting that different ranking criteria retrieve complementary subsets of relevant latents.

These findings motivate combining ranking mechanisms, as even the best individual ranking recovers far fewer relevant latents than the union set. This indicates that each mechanism misses a large fraction of available relevant latents. We also evaluate simple ensembles formed by taking the union of the top-50 latents from two mechanisms.

The ensemble results in Figure 5 show substantial gains for many pairings. Combining Power Score with small-to-intermediate α values ($\alpha \in \{0.1, 0.2, 0.5\}$) with other strong mechanisms often increases the number of relevant latents into the low-to-mid 30s. For instance, the union of Power Score at α=0.01 with α=0.2 yields 36.2 relevant latents on average, while pairing Power Score (α=0.1) with Additive yields 32.0. Frequency Only is particularly complementary. It achieves strong results on its own (25.0) and boosts several Power Score settings (e.g., 34.8 for Power α=0.5 + Frequency Only). Across the grid, the best-performing unions reach roughly 36 relevant latents on average, improving over the best single mechanism by about 8–9 latents, though still leaving a gap to the full union (49.5). Overall, these results support using a small ensemble of diverse ranking criteria to build a richer candidate pool before the final LLM-judge selection.

Steerability via a compute-free proxy. The semantic relevance analysis above captures whether a ranking mechanism retrieves latents whose promoted-token profiles match the target concept. However, concept relevance alone does not guarantee that a latent can be used effectively for steering. Some latents require large steering strengths before their signatures manifest in generation, and may cause text fluency degradation first. We therefore complement the relevance analysis with a second comparison along the steerability axis using our *purity* proxy.

We define a latent's purity as the average logit, under the logit-lens projection, of its top-k promoted tokens. Intuitively, higher purity indicates that a latent has a stronger token-level signature, and is therefore more likely to robustly manifest in generation when used as a steering direction.

Consider this illustrative example of two latents that our LLM judge classifies as semantically rel-

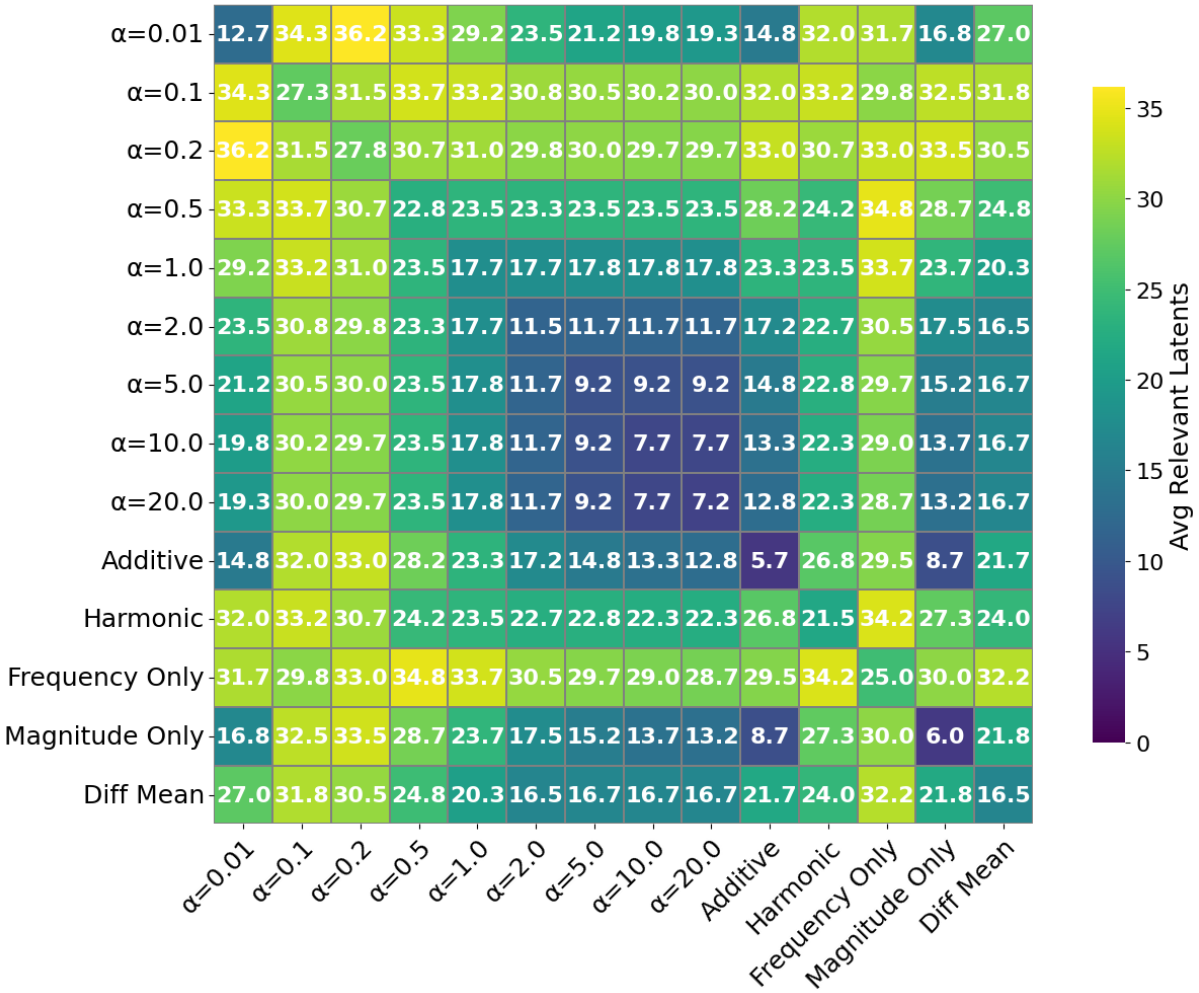

Figure 5: Average number of concept-relevant latents (across the six Ekman emotions) retrieved by taking the union of the top-50 latents from two ranking mechanisms. Diagonal entries correspond to the single-mechanism results.

evant for the concept *joy*, where the latents' logit-lens logit values differ greatly. Latent 1 promotes the top-5 tokens *Appreciate*, *appreciated*, *Hopefully*, *Appreciate*, and *appreciated*, with an average promoted-token logit of 18.3. Latent 2, in contrast, promotes *celebration*, *celebrate*, *celebratory*, *celebrations*, and *celebrating*, with a substantially higher average logit of 100 (i.e., more than $5\times$ larger).

We then steer the model using each latent's decoder direction and compare generations for the same prompt (*"Tell me a story"*). For each latent, we increase the steering-strength coefficient until either (i) the promoted token set is induced or (ii) language degeneration sets in. The contrast is visible in the resulting generations:

Prompt: *Tell me a story*.

Latent 1 (low purity / low promoted-token logits). *"The wind it was a lonely that I would be, for I was very much a creature of the light. But I had been working this will for I was truly looking forward to the light, I couldn' very that now, the means of any kind of means it will be quick and you' really This would be"*

Latent 2 (high purity / high promoted-token logits). *"The old woman, Elara, sat on her porch, the scent of salt and victory dancing in the air. Her weathered face beamed with pride as she watched the sun melt into the ocean, painting the sky in hues of joy and triumph. The celebration had just ended – her village had finally vanquished the Shadow Beast that"*

Qualitatively, the low-purity latent exhibits a clear fluency decay (repetitions, grammatical breakdown) before any reliably joy-related content is

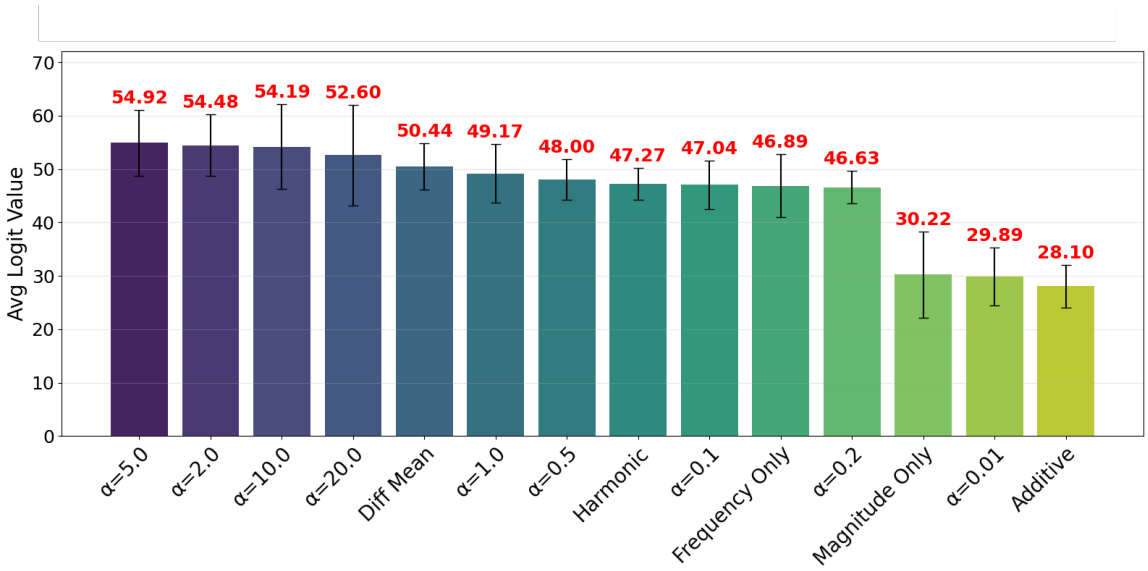

Figure 6: Average latent purity of concept-relevant latents, measured as the mean logit of each latent's top-k promoted tokens under the logit-lens projection, aggregated across the six Ekman emotions. Error bars indicate variation across emotions.

induced, suggesting that increasing λ first destabilizes generation rather than amplifying the intended token-level signature. In contrast, the high-purity latent steers the model towards a coherent, joy-themed continuation in which the latent's token signature also begins to show (e.g., the appearance of celebration-related words), without comparable degeneration at similar steering strengths. We see this behavior in multiple preliminary experiments, motivating purity as a proxy for steerability that complements semantic validation.

The purity analysis in Figure 6 reports the average purity of the latents that were marked as concept-relevant by our semantic validation step. Within our Power Score family, we observe a clear shift between *coverage*, i.e. the number of semantically relevant latents found, and *purity*. The settings that retrieve the most semantically relevant latents (notably $\alpha \in \{0.1, 0.2\}$) tend to have comparatively weaker promoted-token logits (around 47), whereas larger α values yield fewer but substantially higher-purity latents. In particular, $\alpha \in \{2, 5, 10, 20\}$ consistently achieves average promoted-token logits above 50 (peaking around 54–55), indicating stronger and sharper token-level signatures that are more likely to translate into steerability.

Beyond Power Score, DiffMean also produces relatively high-purity latents (50.4), while Harmonic and Frequency Only fall into a mid-range around 47. In contrast, the methods that perform poorly in terms of semantic relevance also tend to yield substantially lower purity. Magnitude Only and Additive achieve average logits of roughly 30 and 28, respectively.

Implications. Taken together, these results highlight a practical trade-off, most clearly within the Power Score family. Smaller-to-intermediate α values tend to maximize *semantic relevance* (coverage of concept-relevant latents), whereas larger α values favor higher *purity* (higher token-level logit values) at the cost of retrieving fewer relevant candidates. We do not observe a single α setting that simultaneously optimizes both axes. This is why our method does not motivate the ranking step as a search for one universally best score. Instead, the ranking stage is designed to create a strong candidate pool for the later selection stage. In practice, this supports either a compromise setting such as α=1 for single-mechanism runs, or a small ensemble of complementary rankings when broader coverage is desired, potentially hedging against noisy or small concept datasets.

A.2 Further Hyperparameters

Beyond the ranking-mechanism choice analyzed above, our pipeline contains a small set of additional hyperparameters and design choices. First, we fix the candidate cutoff at k=50 per ranking. The corresponding analysis from Figure 7 shows that, across the six Ekman emotions, relevant la-

tents keep accumulating roughly linearly as we increase the top-k threshold, rather than saturating quickly after a small number of highest-ranked latents. We therefore treat $k{=}50$ as a pragmatic trade-off motivated by preliminary results. It is large enough to recover a substantial fraction of relevant latents, yet small enough that the resulting candidate pool remains comprehensible for the downstream LLM judge. When we use multiple rankings (or multiple Power-score exponents), we form the final pool by taking the union of the corresponding top-k sets.

Second, for the logit-lens interpretation step we extract the top-$k_{\text{tok}}{=}5$ promoted tokens per latent and summarize their strength by the average promoted-token logit defined in Section 3.4. The steering example above highlights that a small promoted-token set already captures the characteristic lexical "signature" of a latent and makes contrasts in sharpness and purity easy to diagnose, motivating $k_{\text{tok}}{=}5$ as a compact but informative default.

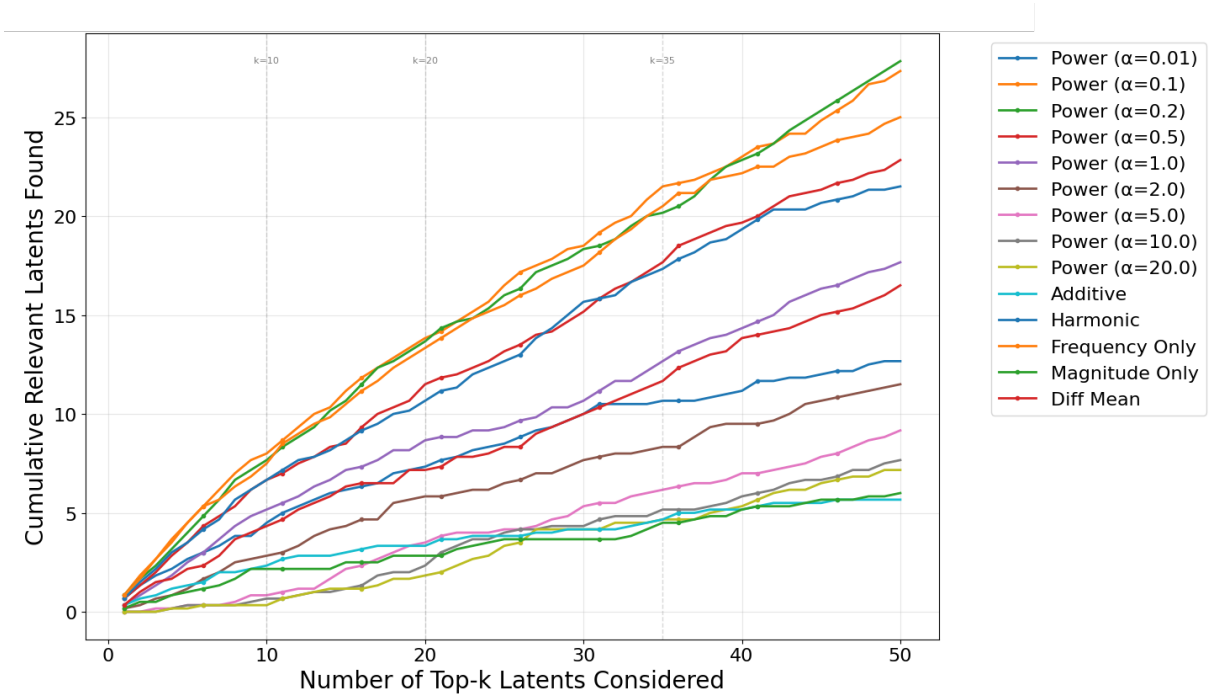

Figure 7: Cumulative discovery of concept-relevant latents as a function of the top-k cutoff, aggregated across the six Ekman emotions. Each curve corresponds to a ranking mechanism (and, for Power Score, a choice of α) and reports how many latents among the first k ranked candidates are judged concept-relevant based on their promoted-token profiles. The roughly linear growth indicates that relevant latents are not confined to the very top ranks, motivating a moderately large cutoff (k=50) as a coverage–comprehensibility trade-off for downstream LLM judge selection.

B LLM Judge

In this appendix, we provide our LLM judge prompt for selecting SAE latents based on their promoted token profile. Throughout all experiments, we use GPT-5.2 as our LLM judge. We set the reasoning parameter to *high* and the text verbosity parameter to *low*.

Semantic-Validation Judge Prompt

```
You are identifying the foundational SAE
↪  neurons that best represent the
↪  concept: '{target_concept}'.

**Goal**: Construct a precise steering
↪  vector by selecting only the neurons
↪  that capture the essence of the
↪  concept, filtering out diluted or
↪  circumstantial associations.

**Selection Guidelines**:
1. **Prioritize Exact Matches**: Retain
↪  neurons where the predicted tokens are
↪  direct synonyms, exact references, or
↪  the concept itself.
2. **Exclude Contextual & Compositional
↪  Features**: Do not include neurons
↪  that represent:
   - *Contextual associations* (e.g., for
   ↪  "Apple", exclude "pie", "tree",
   ↪  "eating").
   - *Grammatical or Structural patterns*
   ↪  (e.g., typical adjectives,
   ↪  prepositions, or sentence
   ↪  structures).
   - *Related but distinct* concepts.
3. **Maintain Purity**: Omit neurons that
↪  appear vague or polysemantic.
4. **Avoid Redundancy**: Before selecting
↪  a neuron to keep:
   - Track which token sets you've already
   ↪  included.
   - Check if any previously selected
   ↪  neuron already has a very similar
   ↪  or nearly identical token set. If a
   ↪  duplicate is found, **EXCLUDE** it
   ↪  to avoid diluting the steering
   ↪  vector with redundant information.

**Output Instructions**:
1. **Re-Rank by Relevance**: The input
↪  list is sorted statistically. Please
↪  output a new list re-ordered by
↪  **semantic relevance**, placing the
↪  most accurate representation first.
2. **Stop at Saturation**: Do not aim for
↪  a specific number of features. If a
↪  small set of latents (even just 1 or
↪  2) fully embodies the concept,
↪  **STOP**. Including lower-quality or
↪  tangential latents may degrade the
↪  steering vector.

**Candidates**:
{candidates_str}

Output Format:
Thinking: [For each latent: brief
↪  reasoning on the selection and
↪  re-ordering, noting any redundancies
↪  detected]
FINAL_LIST: [best_id, second_best_id, ...]
```

C Emotion Steering

Similar to Diallo et al. (2025), we conduct our emotion steering evaluation using a collection of 19 diverse prompts, which we list below for completeness.

1. *"Come up with a fairy tale."*
2. *"Comment on a review of a business written by a customer."*
3. *"Compare the cultural value of theatres and cinemas."*
4. *"Compare the quality of coffee and tea."*
5. *"Describe a fictional character."*
6. *"Describe an idea of what a fulfilling and meaningful life should be like."*
7. *"Describe a piece of art."*
8. *"Describe a typical day for a family in the 1980s."*
9. *"Describe an imaginary board meeting at which the concept of PopTarts is proposed."*
10. *"Pick a colour. Write about that colour as if it was a person."*
11. *"Rewrite a famous film scene from the perspective of a fish."*
12. *"Summarise the plot of 'The Lord of the Rings' by J.R.R. Tolkien."*
13. *"Write a couple of lyrics about society."*
14. *"Write a letter to your parents."*
15. *"Write a scene between a mentor and a mentee in which the mentee must stand up for himself or herself because the mentor doesn't give him or her enough credit."*
16. *"Write a story about an encounter of a boy and a girl."*

17. *"Write a story that involves magic."*

18. *"Write a story that starts with 'The moon was bright. Not that the person could see it."*

19. *"Write an opinion on the role of government in society."*

D Artifact Use

We use existing public research artifacts only and do not collect new data. The main artifacts are GoEmotions for English emotion-labeled Reddit comments, AxBench/Concept500 for broad concept-steering evaluation, AlpacaEval prompts for instruction-following evaluation, Gemma 2 9B-IT, Llama 3.1 8B-Instruct, Gemma Scope SAE, BatchTopK SAE, and pretrained automatic evaluators. We cite the creators of these artifacts in the main text. We use all artifacts only for research evaluation and do not redistribute model weights, benchmark data, or raw dataset examples.

GoEmotions is used only through the released dataset and labels. The emotion-steering subset is English and its class counts are reported in Table 1. AxBench/Concept500 contains 500 concepts, with 72 positive and 72 negative instruction–response samples per concept for steering-vector construction. Emotion evaluation uses 19 held-out prompts listed in Appendix C. For reproducibility, our released code will specify exact checkpoints, dataset versions, licenses, and model-access conditions.

Where available, we verified and followed the corresponding artifact licenses or terms of use. AxBench is released under Apache-2.0, Gemma Scope SAE weights under CC-BY-4.0, the BatchTopK Llama SAE checkpoint used in our experiments under Apache-2.0, Gemma models under the Gemma Terms of Use, and Llama 3.1 models under the Llama 3.1 Community License.