Multi-Concept Steering in Large Language Models (LLMs)

Anonymous ACL submission

Abstract

Large Language Models (LLMs) excel at producing fluent text yet remain prone to generating harmful or biased outputs, largely due to their opaque, "black-box" nature. Existing mitigation strategies, such as reinforcement learning from human feedback (RLHF) and instruction tuning, can reduce these risks but often demand extensive retraining and may not generalize. An alternative approach leverages sparse autoencoders (SAEs) to extract disentangled, interpretable representations from LLM activations, enabling the detection of specific semantic attributes without modifying the base model. In this work, we extend the Sparse Conditioned Autoencoder (SCAR) framework (Härle et al., 2024) to enable multi-attribute detection and steering. Our approach disentangles multiple semantic features—such as toxicity and style—in a unified latent space, providing granular, real-time control without compromising textual quality. Experimental results demonstrate that our multi-feature extension maintains the interpretability, safety, and quality of the original single-attribute SCAR while offering enhanced flexibility by allowing simultaneous control over multiple semantic attributes. Furthermore, evaluations under both black-box and white-box adversarial attack scenarios reveal that our approach remains robust, reinforcing its potential as a reliable and adaptable safety mechanism for LLMs.

1 Introduction

Large language models (LLMs) have revolutionized natural language processing by generating fluent and contextually appropriate text. However, their "black-box" nature often makes them susceptible to producing toxic, biased, or otherwise harmful content—a risk that has spurred a rich line of research into methods for controlling undesirable behavior.

In this work, we extend the SCAR framework by developing a multi-feature latent conditioning approach that enables simultaneous detection and steering of several key semantic attributes. Our contributions are threefold: (1) We propose a novel training strategy that conditions multiple latent dimensions in a pre-trained LLM using sparse autoencoders and disentanglement constraints; (2) We demonstrate through extensive experiments that our method achieves flexible, fine-grained control over diverse attributes without compromising overall generation quality; and (3) We demonstrate the robustness of our multi-feature steered model against adversarial jailbreak attacks in both white-box and black-box settings. This work thus represents a significant step toward more controllable, interpretable, and safer large language models.

2 Related Work

Early approaches to mitigating these risks have primarily focused on techniques such as reinforcement learning from human feedback (RLHF) and instruction tuning. For instance, RLHF methods (Ziegler et al., 2020; Ouyang et al., 2022) fine-tune models using human-annotated data so that their outputs better align with ethical and performance standards, while instruction tuning trains models on datasets enriched with explicit directives. Although effective in many cases, these methods require extensive retraining or fine-tuning, are computationally expensive, and tend to impose static guardrails that may not generalize well to diverse contexts.

An alternative and promising strategy leverages sparse autoencoders (SAEs) to extract interpretable, disentangled representations from LLM activations. SAEs work by reconstructing high-dimensional activation vectors using a low-dimensional, sparsely activated code. This sparse coding approach not only yields representations that are more interpretable—as demonstrated by Gao et al. (2024) and Cunningham et al. (2023)—but also provides a mechanism for detecting specific semantic at-

tributes (e.g., toxicity) without altering the original language model.

Building on these ideas, SCAR (Sparse Conditioned Autoencoders for Concept Detection and Steering in LLMs) introduced a latent conditioning mechanism that forces a designated latent dimension to align with a target concept, enabling real-time detection and steering of harmful outputs during inference (Härle et al., 2024). Although SCAR presents an elegant solution for single-attribute control, many practical applications require simultaneous regulation of multiple features—such as toxicity, sentiment, and stylistic attributes—to achieve a more comprehensive safety and quality standard.

3 Sparse Auto Encoder Integration

3.1 Architecture

We follow the SCAR framework, inserting a Sparse Autoencoder into the LLM pipeline at block D. Specifically, for each token's hidden-state activation $\mathbf{x}$, we compute:

$$\mathbf{h} = W_{\text{enc}}\,\mathbf{x} + \mathbf{b}_{\text{enc}}. \tag{1}$$

Here, W_{enc} and $\mathbf{b}_{\text{enc}}$ are the parameters of the SAE's encoder. We then apply σ, a sparse activation function, to $\mathbf{h}$:

$$\mathbf{f} = \sigma(\mathbf{h}), \tag{2}$$

yielding the activated latent representation. We pass $\mathbf{f}$ into the decoder,

$$\bar{\mathbf{x}} = W_{\text{dec}}\,\mathbf{f} + \mathbf{b}_{\text{dec}}, \tag{3}$$

where W_{dec} and $\mathbf{b}_{\text{dec}}$ are the decoder parameters. During training, the decoder output $\bar{x}$ is not used by the LLM; instead, the original x is used for final predictions in the dataset creation phase. During inference, however, we replace x with $\bar{x}$ in the LLM, enabling the model's output to be steered via the latent representation.

3.2 Multi-Feature Conditioning

To jointly learn multiple features, we designate m latent dimensions in h, each corresponding to a feature j. Formally:

$$h_j \leftrightarrow \text{feature } j, \text{ for } j = 1, \ldots, m.$$

These dedicated neurons are constrained to align with each feature's label y_j by a condition loss (detailed below). The remaining latent dimensions of h remain unconstrained and serve for reconstruction.

4 Training Objective

4.1 Reconstructed Loss

We encourage the SAE to approximate the original activation x via:

$$L_{recon} = \frac{(||\mathbf{x}-\bar{\mathbf{x}}||^2)}{(||\mathbf{x}||^2)},$$

Where $\bar{x} = W_{dec}\ f + b_{dec}$. The normalization by $||x||^2$ stabilizes training across a wide range of activation magnitudes.

4.2 Multi-Feature Conditioned Loss

To ensure each feature dimension h_j correlates with its label y_j, we apply a binary cross-entropy (BCE) on the logit h_j. Concretely:

$$L_{cond} = \Sigma_{j=1}^{m}\ \text{BCE}(\sigma(h_j), y_j)$$

where σ is the standard sigmoid function. If a label is unknown $y_j = -1$, we mask out that term. This loss drives neuron h_j to be large and positive when $y_j \approx 1$ and near zero when $y_j \approx 0$.

4.3 Combined Objective

We freeze all LLM parameters and only optimize SAE parameters W_{enc}, b_{enc} . The overall training loss is,

$$L_{total} = L_{recon} + \lambda L_{cond}$$

where λ balances feature conditioning against reconstruction fidelity. We train using stochastic gradient descent (e.g., Adam) over the dataset's token-level activations.

5 Inference Time Steering

To steer our features, we take our h_j latent vector and scale it by some hyperparameter value α_j or keep it the same otherwise. We express this new activation f_j as:

$$f_j = \begin{cases} \alpha_j \cdot h_j & \text{if } j \text{ corresponds to a concept} \\ \sigma(h_j) & \text{otherwise} \end{cases}$$

Where j corresponds to the jth feature we want to steer and α_j is the scaler value we apply to our h_j feature. We then take this new latent vector, decode it, and add it to the Feed Forward section of our transformer block.

6 Experiment Setup

To evaluate the Multi-Concept SCAR framework, we conducted a series of experiments leveraging the LLaMA-8B model, focusing on its ability to simultaneously represent and interact with multiple concepts. The experimental design builds on the methodology outlined in the SCAR paper, adapting it to a novel combination of datasets and concept analyses.

To evaluate the effectiveness of our multi-feature Sparse Autoencoder (SAE) within the SCAR framework, we conduct a comparative analysis of SAE reconstruction loss across different architectures. Specifically, we compare the SAE loss of our multi-feature SCAR model against the original SCAR model, which leverages sparse encoding but does not incorporate multiple contextual dependencies. By analyzing SAE loss, we assess the extent to which each model effectively reconstructs its learned representations, providing insight into the benefits of incorporating multiple features.

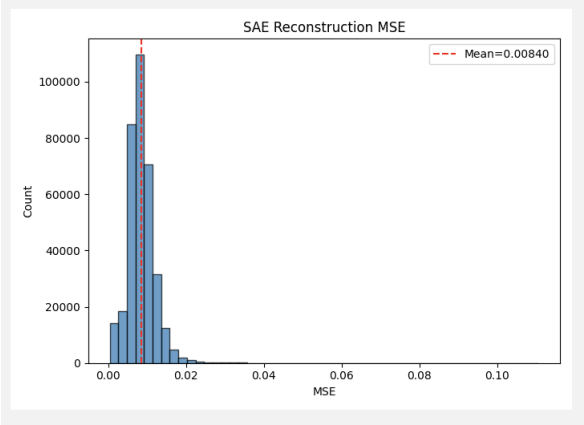

Figure 1: SAE Reconstruction Loss for M-SCAR

In Figure 1, we compare the original activations from our LLaMA-8B model to those reconstructed by M-SCAR. The consistently low SAE loss indicates that the sparse autoencoder imposes minimal distortion on the underlying representations, effectively preserving the model's capacity and behavior while enabling multi-concept steering.

To further evaluate the efficacy of our approach for concept detection, in Figure 2, we plot the concept score on the y-axis against the test sentence index on the x-axis, where each index corresponds to an individual sentence from the set:

Sentences

⟨"Thou art a knave and thou shall speak no more!"
"I will kill you, you worthless piece of trash!"
"And now, dear friend, let us enjoy a pleasant meal."
"But soft, what light through yonder window breaks?"
"You're an idiot. I hope you suffer!"
"I love the morning sky, it's so peaceful."⟩

In our experiments, we focus on two target attributes: toxicity and Shakespearean style. The evaluation graphs reveal that sentences exhibiting high toxicity consistently yield elevated toxicity scores, while those lacking toxic language score low. Similarly, sentences written in a Shakespearean style register high concept scores for that attribute, in contrast to modern or neutral sentences. This clear separation across the test samples confirms that the SAE not only reconstructs the original activations faithfully but also effectively disentangles and isolates the semantic features required for robust concept detection. These results validate that our multi-feature SCAR extension successfully extracts actionable latent signals, forming a robust basis for downstream steering.

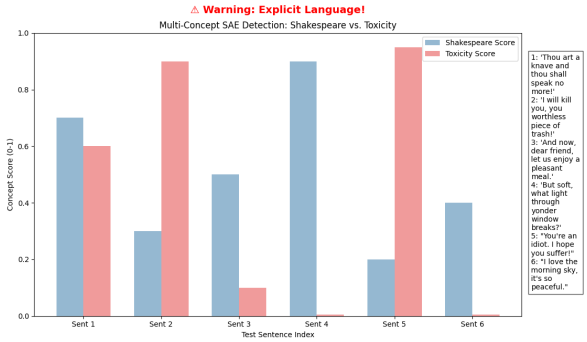

Figure 2: A bar graph showing, for each sentence index (sentences are shown in the right-hand side box), two adjacent concept scores representing the Shakespearean style and the toxicity style scores, respectively.

As jailbreak attacks on LLMs become increasingly sophisticated, assessing model robustness against these adversarial techniques is essential. Recent works (Zou et al., 2023; Chao et al., 2024) have demonstrated that aligned language models are vulnerable to prompt injections, adversarial rephrasings, and automated attack strategies that can bypass safety mechanisms. We further investigate whether our M-SCAR model can curb the impact of state-of-the-art jailbreaking methods. We evaluated the resilience of our model using a set of benchmarked adversarial prompting techniques from (Zou et al., 2023; Chao et al., 2024). For

each method, we measured the model's toxicity score under both normal and jailbroken conditions, comparing our M-SCAR model against a baseline model.

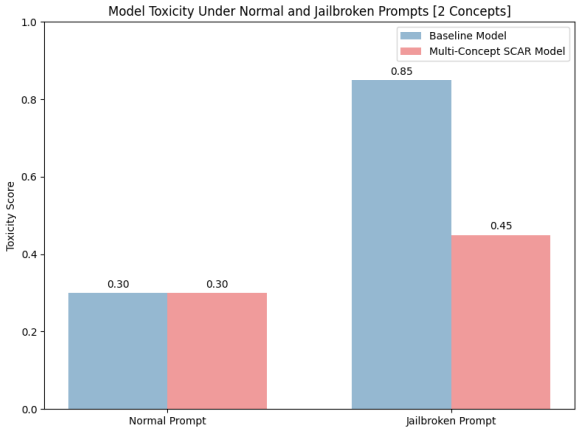

Figure 3: Model toxicity under normal and jailbroken prompts for two concepts. The baseline model exhibits a significant increase in toxicity under adversarial prompting (0.85), whereas M-SCAR reduces this to 0.45, demonstrating its robustness.

Figure 3 presents the results of our jailbreak evaluation. The baseline model exhibits a drastic increase in toxicity when subjected to adversarial prompts, highlighting its vulnerability to SOTA jailbreaking techniques. In contrast, our M-SCAR model successfully mitigates these effects, reducing the toxicity score from 0.85 to 0.45 under jailbreak conditions. These results suggest that multi-concept latent steering via SAEs provides a promising direction for enhancing LLM robustness against adversarial attacks.

7 Conclusion / Limitations

In this paper, we introduced a multi-concept extension of the Sparse Conditioned Autoencoder (SCAR) framework, enabling simultaneous detection and steering of multiple semantic attributes within LLMs. Unlike traditional methods such as RLHF and instruction tuning—which often require expensive model retraining—our approach minimally modifies existing LLMs by inserting and training an external sparse autoencoder layer. This architecture preserves the core model's expressive capacity while disentangling key semantic dimensions (like toxicity and style) into interpretable latent features that can be dynamically scaled or suppressed during inference. Our extensive experiments demonstrated that this multi-feature design not only preserves text quality and maintains fine-grained control, but also introduces strong resilience against both white-box and black-box adversarial "jailbreak" prompts.

Although our experiments focused on disentangling and steering two attributes (toxicity and style), the multi-concept SCAR framework naturally extends to higher-dimensional settings. We envision several directions for future research. First, a systematic assessment is needed to determine the upper bound on the number of concepts (n) that can be reliably disentangled before sparse autoencoder performance degrades. Characterizing where (and why) such a falloff occurs will inform architectural and training enhancements for handling a larger palette of user-defined attributes. Second, more sophisticated adversarial methodologies can be incorporated to further stress-test the model's robustness, including automated or adaptive attack strategies that target multiple latent dimensions simultaneously. Finally, integrating multi-concept SCAR with complementary alignment techniques—such as selective fine-tuning or reward-modeling—may offer an even stronger composite defense against both accidental and adversarial misuse of LLMs.

References

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. 2024. Jailbreaking black box large language models in twenty queries. *Preprint*, arXiv:2310.08419.

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

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. *Preprint*, arXiv:2406.04093.

Ruben Härle, Felix Friedrich, Manuel Brack, Björn Deiseroth, Patrick Schramowski, and Kristian Kersting. 2024. Scar: Sparse conditioned autoencoders for concept detection and steering in llms. *Preprint*, arXiv:2411.07122.

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. *Preprint*, arXiv:2203.02155.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Chris-

tiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences. *Preprint*, arXiv:1909.08593.

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

A SCAR MSE Loss and Delta Loss

In this appendix, we provide additional details on the loss functions employed in the SCAR framework, namely the Mean Squared Error (MSE) loss used for activation reconstruction and the Delta loss that enforces latent space disentanglement.

A.1 SCAR MSE Loss

The reconstruction loss in SCAR is computed as the Mean Squared Error (MSE) between the original activations a and the reconstructed activations $\hat{a}$. Formally, the loss is defined as:

$$L_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^{N} (a_i - \hat{a}_i)^2, \tag{4}$$

where N is the number of activation elements. This loss encourages the sparse autoencoder to capture and accurately reconstruct the high-dimensional activations using a low-dimensional, sparsely activated code. The accurate reconstruction of these activations is critical for preserving the semantic information necessary for downstream concept detection and steering.

In our experiments, we evaluated the reconstruction performance over 10 batches and obtained an average MSE of approximately 0.0015. This low error indicates that the model is able to effectively reconstruct the activations while maintaining the semantic fidelity required for concept detection.

A.2 Discussion

In our experiments comparing the multi-feature SCAR with its single-feature counterpart, we observed that the multi-feature model exhibits a higher reconstruction MSE. This increased loss is not a shortcoming, but rather an expected trade-off: while the single-feature SCAR can focus on accurately reconstructing activations for one specific attribute, our multi-feature approach must capture a richer and more complex representation to support simultaneous detection and control of multiple semantic attributes. Thus, the higher reconstruction loss in the multi-feature SCAR reflects the additional complexity required to manage several features at once, and is a necessary cost for achieving more comprehensive and flexible control. Future work will explore methods to mitigate this increased error while retaining the benefits of multi-attribute handling.

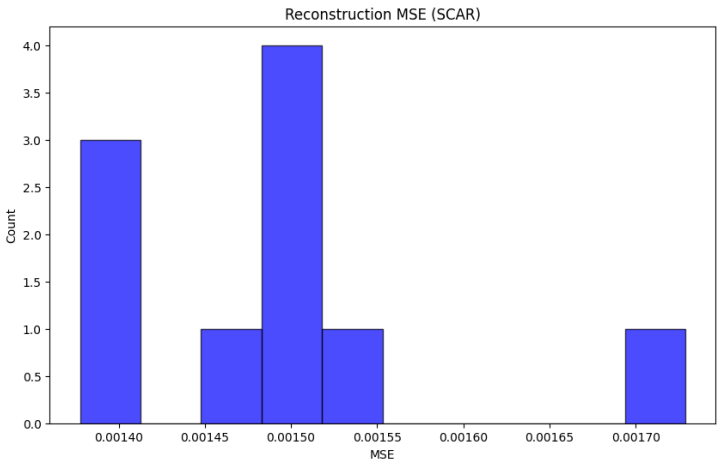

Figure 4: Reconstruction MSE across 10 batches, showing an average MSE of approximately 0.0015.