Evaluation of SAE-based Refusal Features in Base and Chat LLMs: Dataset-specific Characteristics for SAEs

Master Thesis Proposal

Tilman Kerl
Technical University of Vienna
Department of Informatics and Data Science

Supervisors:

Prof. Dr. Peter Knees (Main Supervisor)
Ilya Lasy, MSc (Second Supervisor)

December 16, 2024

1 Problem Statement

A key challenge in safely deploying large language models (LLMs) lies in ensuring that they consistently refuse to respond to unsafe prompts while still engaging constructively with safe ones [2].

Although fine-tuning on curated datasets has been the predominant solution for achieving such refusal behaviour, this approach often struggles to generalize beyond its training distribution, especially when facing adversarial or multi-turn prompts. To address these limitations and foster understanding of the workings of these models, recent research has begun exploring methods that intervene on the model's activations at inference time - referred to as feature steering - rather than relying solely on weight updates through fine-tuning [12, 8, 7, 13].

An emerging method for feature steering leverages sparse autoencoders (SAEs) [13, 2] trained on a model's activations. SAEs aim to discover a more interpretable, disentangled representation of the neural computations underlying a model's behaviour. Central to this effort is the superposition hypothesis: large neural networks often pack a vast number of distinct features into a limited set of neurons by representing them as overlapping vectors in a high-dimensional space. This compression, known as polysemanticity, can cause a single neuron or dimension to encode multiple, conceptually unrelated features. While polysemantic encoding efficiently utilizes available resources, it hinders mechanistic interpretability — making it challenging to isolate the features that mediate specific behaviours, such as refusals.

SAEs provide a potential solution by learning a sparse, over-complete basis for the model's hidden representations. In doing so, they strive to "decompress" this polysemantic tangle into more monosemantic features - i.e., more directly interpretable latent dimensions that correspond to distinct, coherent concepts or behaviours.

By training an SAE on the model's activations, we can identify latent features that strongly correlate with refusal behaviour. Once identified, these features can either be manipulated at inference time by "steering" them - clamping or shifting their activation values - to encourage or discourage certain responses without retraining the model. Alternatively, these feature directions can be employed directly in the model by multiplying their values to the respective part of the model.

This approach complements existing safety methods, offering the potential to improve generalisation and adapt while leaving the model's underlying parameters untouched.

2 Goals and Expected Outcome

- Investigate SAEs data-set dependency and performance on refusal task.
- What dataset characteristics promote strong refusal features?
- Contribution to (fully) open SAE research
- Fully open SAE

3 Research Questions

The following section presents our research questions and explains them.

RQ 1: How does the choice of training dataset (e.g., pre-training corpus vs. chat-refinement data) affect the sparse autoencoder's ability to isolate and represent "refusal" features within a language model's latent activations?
LLMs and chat-tuned models often lack transparency about their training data, hindering analysis of how datasets shape safety-critical behaviors like refusals. Fully open-source models provide a unique opportunity to explore this through sparse autoencoders (SAEs), which can isolate and manipulate refusal-related features without altering model weights
RQ 2: Which characteristics of the underlying training data (e.g., distributional properties, topical diversity, presence of refusals) are most predictive of the strength and clarity of SAE-extracted refusal features?

RQ 3: How do SAEs trained on chat-oriented datasets compare to those trained on the original pre-training corpus in terms of robustness and interpretability of refusal-related features, and what is their impact on downstream controllability in refusal tasks?

4 Research Methods

4.1 Models and Configurations

At the core of our experiemtns are the Pythia model family [5] which was specifically designed to be analyesed and has it's entire code [5, 4, 9] and data [3] open source. Here we look at the different sizes (from 410m - to 2.8b). Firstly, we need to extract which model is the first one to exhibit a refusal behaviour that can properly be evaluated. E.g. pythia 70m often just repeats the prompt or produces gibberish.

Further models we might analyse is the e.g. OLMo model [10] which is also a fully open model .

4.2 Training SAEs

Pre-Training SAE. Train SAEs on activations from the base - model (pre-training) dataset.
Post-Training SAE. Train SAEs on activations derived from a post-training dataset.

4.3 Refusal Feature Extraction

Following Ardeiti or Microsoft or Nanda we extract the refusual direction.

5 Evaluation

The evaluation is structured into three distinct phases, following well-established and proven methodologies and metrics (see [7] for reference).

5.1 Phase 1: SAE Quality Metrics

This phase focuses on evaluating the SAEs independently to verify their effectiveness in reconstructing and sparsely encoding model activations.

5.1.1 Reconstruction Error

We measure how accurately each SAE reconstructs the original activation patterns of the language models. Reconstruction quality will be quantified by minimizing the loss between input activations and their SAE-generated reconstructions.

5.1.2 Sparsity

To ensure and evaluate how much the extracted features remain interpretable and disentangled, we calculate the fraction of non-zero activations in the sparse vector representation produced by the SAEs. Sparsity metrics will follow prior work, as described by Arditi et al. [1].

5.2 Phase 2: Refusal

This phase investigates how refusal-related features, extracted via SAEs, influence the models' refusal behavior across safe and unsafe instructions. By enabling or disabling SAE-informed refusal features, we can measure their direct impact on refusal and over-refusal rates, isolating the effect of these features on model behavior.

5.2.1 Refusal Rate

We evaluate the models' ability to refuse unsafe prompts, measuring refusal frequency using benchmarks like Wild Guard [**?**] and JailbreakBench [6]. Refusal will be detected by identifying common refusal phrases in model completions, following the methodology of Kissane et al. [7].

5.2.2 Over-refusal Rate

To ensure the refusal features do not inadvertently reduce the models' compliance with benign prompts, we measure the over-refusal rate for safe instructions. This evaluation compares model outputs for paired safe and unsafe instructions, as in Kissane et al. [7].

5.3 Phase 3: Robustness and Downstream Performance

In this phase, we assess the broader impact of SAE-informed refusal features on model robustness, reasoning, and general performance across tasks. Comparisons will be conducted across four configurations: the base model, the chat-tuned model, and their respective SAE-enhanced counterparts. We explicitly disable identified refusal features or use neutral feature clamping (e.g., setting activations to zero or a baseline value) to quantify their specific contributions to task performance and refusal robustness. These ablations help evaluate whether refusal features interfere with unrelated capabilities or improve safety alignment.

5.3.1 Massive Multitask Language Understanding (MMLU)

We employ the MMLU benchmark [11] to evaluate the models' overall reasoning and performance across diverse tasks. This analysis focuses on determining how the integration of refusal features impacts capabilities unrelated to refusals.

5.3.2 Jailbreak Robustness

We test model robustness against adversarial prompts designed to bypass refusal mechanisms using benchmarks like JailbreakBench [6]. Multi-turn evaluation setups, similar to those in [7], will be used to analyze the models' resilience to jailbreak attempts and the effectiveness of SAE-informed refusal features.

5.3.3 Baseline Comparisons

To establish a clear understanding of the effects introduced by the SAEs, we conduct baseline comparisons with standard models (base and chat-tuned) that do not include SAE-informed modifications. This includes ablation studies to isolate and evaluate the specific contributions of the refusal features:

6 State of the Art

6.1 Sparse Autoencoders and Superposition

6.2 Steering Large Language Models

Concepts to mention:

- Mechanistic Interpretability in LLMs: toy-models and Scaling Interpretability Techniques (Claude)

- Superposition and Feature Representation, Monosemanticity and Polysemanticity

6.3 Refusal of Harmful Content

7 Relevance to the Curriculum

This work tries to explore and uncover the workings of generative decoder transformer models working with natural language. Thus, bringing together various aspects from different courses and core elements of the Data Science and Informatics curriculum. Courses such as `Machine Learning` (188.702), `Natural Language Processing and Information Extraction` (194.093), and `Applied Deep Learning` (194.077) have provided a strong foundation in machine learning techniques, including deep learning architectures central to transformer models. Additionally, `Advanced Information Retrieval` (188.980), which explores state-of-the-art retrieval techniques like BERT-based models, directly supports this research's focus on identifying and refining task-specific circuits within language models.

The course `Research Topics in Natural Language Processing` (194.135) has introduced advanced methodologies specific to NLP, equipping me with essential research skills in e.g. examining model interpretability and ethical behaviour in language models.

Additional expertise in handling complex data systems, as developed and studied in `Advanced Database Systems` (184.780) and `Data-intensive Computing` (194.048), is critical for the large-scale model processing and analysis involved in this work. Furthermore, courses like `Information Visualization` (186.143) and `Visual Data Science` (186.868) provide a solid foundation in visualising complex data, which will be crucial in effectively communicating the results of model circuit analysis and interventions.

8 Overall Review

- Provide a summary of the research proposal's main points.
- Reflect on the expected impact of the research within the field.
- Highlight any broader implications or future research opportunities.

References

[1] Andy Arditi, Oscar Obeso, A. S. D. P. N. P. W. G., and Nanda, N. Refusal in language models is mediated by a single direction, 2024.

[2] Bereska, L., and Gavves, E. Mechanistic interpretability for ai safety – a review, 2024.

[3] Biderman, S., Bicheno, K., and Gao, L. Datasheet for the pile. *arXiv preprint arXiv:2201.07311* (2022).

[4] Biderman, S., Prashanth, U. S., Sutawika, L., Schoelkopf, H., Anthony, Q., Purohit, S., and Raff, E. Emergent and predictable memorization in large language models.

[5] Biderman, S., Schoelkopf, H., Anthony, Q. G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning* (2023), PMLR, pp. 2397–2430.

[6] Chao, P., Debenedetti, E., Robey, A., Andriushchenko, M., Croce, F., Sehwag, V., Dobriban, E., Flammarion, N., Pappas, G. J., Tramèr, F., Hassani, H., and Wong, E. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. In *NeurIPS Datasets and Benchmarks Track* (2024).

[7] Connor Kissane, Robert Krzyzanowski, A. C., and Nanda, N. Base llms refuse too. Alignment Forum, 2024.

[8] Connor Kissane, Robert Krzyzanowski, N. N., and Conmy, A. Saes are highly dataset dependent: A case study on the refusal direction. Alignment Forum, 2024.

[9] Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., Le Noac'h, A., Li, H., McDonell, K., Muennighoff, N., Ociepa, C., Phang, J., Reynolds, L., Schoelkopf, H., Skowron, A., Sutawika, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. A framework for few-shot language model evaluation, Sept. 2021.

[10] Groeneveld, D., Beltagy, I., Walsh, P., Bhagia, A., Kinney, R., Tafjord, O., Jha, A., Ivison, H., Magnusson, I., Wang, Y., Arora, S., Atkinson, D., Authur, R., Chandu, K. R., Cohan, A., Dumas, J., Elazar, Y., Gu, Y., Hessel, J., Khot, T., Merrill, W., Morrison, J. D., Muennighoff, N., Naik, A., Nam, C., Peters, M. E., Pyatkin, V., Ravichander, A., Schwenk, D., Shah, S., Smith, W., Strubell, E., Subramani, N., Wortsman, M., Dasigi, P., Lambert, N., Richardson, K., Zettlemoyer, L., Dodge, J., Lo, K., Soldaini, L., Smith, N. A., and Hajishirzi, H. Olmo: Accelerating the science of language models. *arXiv preprint* (2024).

[11] Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300* (2020).

[12] Joseph Bloom, C. T., and Chanin, D. Saelens. https://github.com/jbloomAus/SAELens, 2024.

[13] Kyle O'Brien, David Majercak, X. F. R. E. J. C. H. N. D. C. E. H., and Poursabzi-Sangde, F. Steering language model refusal with sparse autoencoders, 2024.