Interpretability with Sparse Autoencoders: Literature Review & Towards Multimodality

Name: Eric He
Email: eriche26@stanford.edu
SUNet ID: eriche26

Abstract

The interpretability of machine learning systems has quickly become one of the largest problems in the field. Sparse autoencoders (SAEs) have quickly become a novel and promising route to model interpretability. In this paper, we discuss the work on SAEs since the initial paper in 2023, specifically in vision, and we propose a methodology to move forward with using SAEs for interpretability of modern multimodal Transformer-based models. We also present some preliminary results in using this methodology for a simple vision-language model for image captioning, finding that interpretability persists through post-embedding decoder layers.

1 Introduction

Interpretability of deep machine learning models has become more and more of a pressing issue as models grow exponentially and become more complex. Interpretability is not only a key to ensuring aligned, understandable, and open AI - it is also important to furthering the innovation of cutting-edge models.

> *Despite the extraordinary success of ML and AI in multimedia and other fields which require intelligent processing, the interpretability of ML/AI remains a persistent challenge. Specifically, the black-box nature of contemporary ML architectures has posed a longstanding problem, causing concerns about questionable performances and predictions in real applications.* Gao & Guan (2023)

Most large language models (LLMs) have largely become based on the Transformer architecture, first introduced in Vaswani et al. (2017). The Transformer's modular, layered architecture has lended itself well to interpretability. Initial approaches included attention visualization, which attempted to interpret the integral Attention layers in Transformers (see Xu et al., 2016). Sparse autoencoders were found to be an extremely effective manifestation of this feature, used to solve polysemanticity Cunningham et al. (2023).

Recent LLMs have increasingly become multimodal - capable of consuming or producing multiple types of inputs across textual, visual, and audio contexts. The most basic and most-studied multimodal models are vision-language models (VLMs). However, to the author's knowledge, neuron-level interpretability (such as SAEs) has yet to be studied in these models.

Most modern multimodal VLMs are hybrids of vision and language encoders with language decoders. These encoders and decoders are often parts of existing models that are fine-tuned to fit together, leveraging cross-attention to connect the components (see Rajan et al., 2022).

After the cross-attention layer, the model reaches an output after more Transformer decoder layers. However, the behavior in these layers is very much unstudied with sparse autoencoders.

In this paper, we present a simplified experiment, attaching a sparse autoencoder to a basic vision-language model. We also propose a framework for future evaluations of SAEs on VLMs, which we hope offers a starting point for more work on SAE-based interpretability in not only VLMs, but also other types of multimodal models.

2 Related Work

In the seminal paper on SAEs and interpretability, Cunningham et al. (2023), sparse autoencoders were used to reduce polysemanticity in a language model by encouraging sparsity in higher-dimensional space. The SAE neurons were found to have highly interpretable meanings, and could be labeled by human or by LLM by seeing the corresponding text/tokens. Since the paper's publication, an online database [1] of SAEs and SAE features for many open-source LLMs has been created.

SAEs have been applied to vision by several groups recently, including in Gorton (2024), where SAEs were used in InceptionV1, a **traditional convolutional neural network** to detect curve features otherwise invisible in the convolutional layers. While not necessarily aligned with model interpretability, it 1) demonstrates the application of SAEs to vision models; and 2) demonstrates the ability of SAEs to drastically increase monosemanticity.

Moving from CNNs to Transformers, LessWrong user hugofry applied SAEs to CLIP Vision Transformers and analyzed which images maximally activated neurons in the SAE. Surprisingly, this approach found highly interpretable features almost straight out-of-the-box: neurons specifically corresponding to ticks, fish, goats, and mountains show up with strong monosemanticity and low noise.

A paper that applies the interpretability of SAEs is Rao et al. (2024), where researchers applied an SAE to a CLIP Vision Transformer to create a Concept Bottleneck Network, fully limited by human-interpretable features.

Zhou & Miao (2024) introduced the Disentangled Graph Variational Auto-Encoder (DGVAE) to improve both the performance and interpretability of recommender systems that process multimodal data. Although it's not an SAE, we still found this work worth mentioning in the realm of multimodal interpretability. DGVAE constructs disentangled representations that align user-item interactions with interpretable textual features, enhancing recommendation accuracy and providing clear insights into the factors influencing recommendations.

Another very cool application of SAEs is Abdulaal et al. (2024), where authors applied SAEs to the field of radiology by developing SAE-Rad, a model designed for interpretable radiology report generation. SAE-Rad utilizes SAEs to decompose latent representations from a pre-trained vision transformer into human-interpretable features, addressing issues such as hallucinations and lack of interpretability which are ever-important in medicine.

From the related work, it's important to note that 1) none of the works put the SAE in the Transformer decoder blocks, post-embeddings; and 2) none of the works analyze models with simultaneous text/image input. Our experiment and evaluation framework attempts to start to fill both of these gaps.

3 Experiment

To explore how the text Transformer decoder handles image embeddings, we started with a very small HuggingFace model based on a ViT encoder and a GPT2-Small decoder, built for image captioning (Figure 3). Totaling less than 500M parameters, this model was manageable to work on with just a single Google Colab GPU. The SAE functionality was sourced from the wonderful SAELens library by Charles Bloom.

For our SAE, we used an expansion ratio of 1 to 32 and a standard ReLU activation, which has been the historic precedent used for the residual stream of GPT2. We placed the SAE on the residual stream of layer 9 of 12 in GPT2.

Unfortunately, because the version of GPT2 in the library did not have the cross-attention our model was using to feed the vision embeddings to the decoder, a custom PyTorch hook had to be written to intercept the model's decoder blocks and sandwich in the SAE in the residual stream. Furthermore, we could not use the pretrained SAEs from SAELens because the training of the multimodal model touched the parameters of the GPT2-Small decoder.

[1] https://www.neuronpedia.org

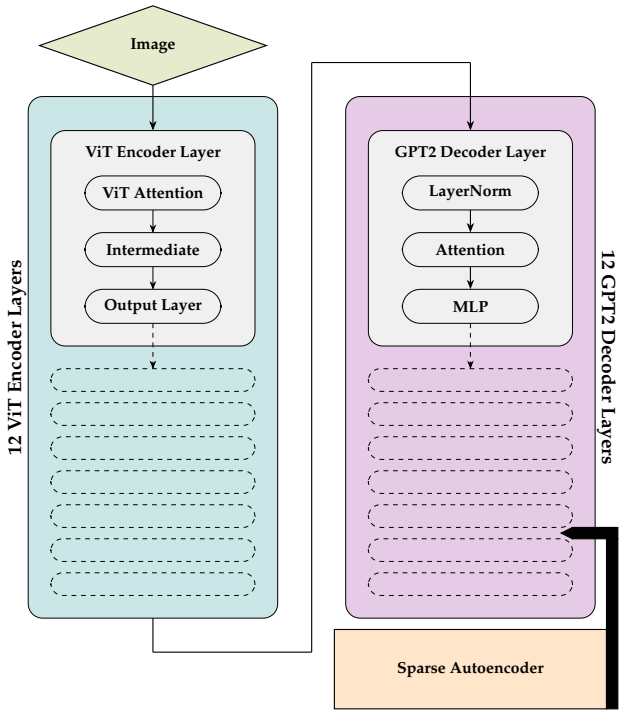

Figure 1: The ViT/GPT2 model used in this paper, with the sparse autoencoder inserted.

We trained the SAE on a small subset (25k images) of the ImageNet-1000 dataset, keeping all other layers of the model frozen. We computed loss only for the SAE:

$$\ell = \underbrace{\left[\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2\right]}_{\text{MSE(original, reconstructed)}} + \lambda \underbrace{\left[\frac{1}{N}\sum_{i=1}^{N} y'_i\right]}_{\text{L1(encoded)}}$$

We trained the model over 5 epochs on the aforementioned subset of ImageNet with $\lambda = 0.001$ and the Adam optimizer with default parameters. This took several hours on a Google Colab L4 GPU, because unfortunately to the author's knowledge, there is an issue with PyTorch hooks and parallelism. After training, we aimed to evaluate the model. This was straightforward - take a random sample of ImageNet, and get the SAE encoded vectors for each image, then much like LessWrong user hugofry's work, we can find the images with the highest activations per neuron.

We ended up taking a sample of around 100,000 images from ImageNet, and created a lightweight Python web application to display the highest-activation images for each neuron. The images needed to be downscaled for practicality with disk space and memory usage constraints.

Figure 2 shows a couple of the interesting neurons we were able to extract. We were surprised to find such clearly monosemantic features this deep in the decoder - if the decoder is converting image embeddings into language, why are there no fish "tanks" in the top images?

These results also mean that we have complete interpretability using SAEs for all stages of VLMs like these: vision transformers with SAEs have proven to be highly effective, and now we have proven that SAEs in the Transformer decoder stage also work well and offer meaningful interpretable features.

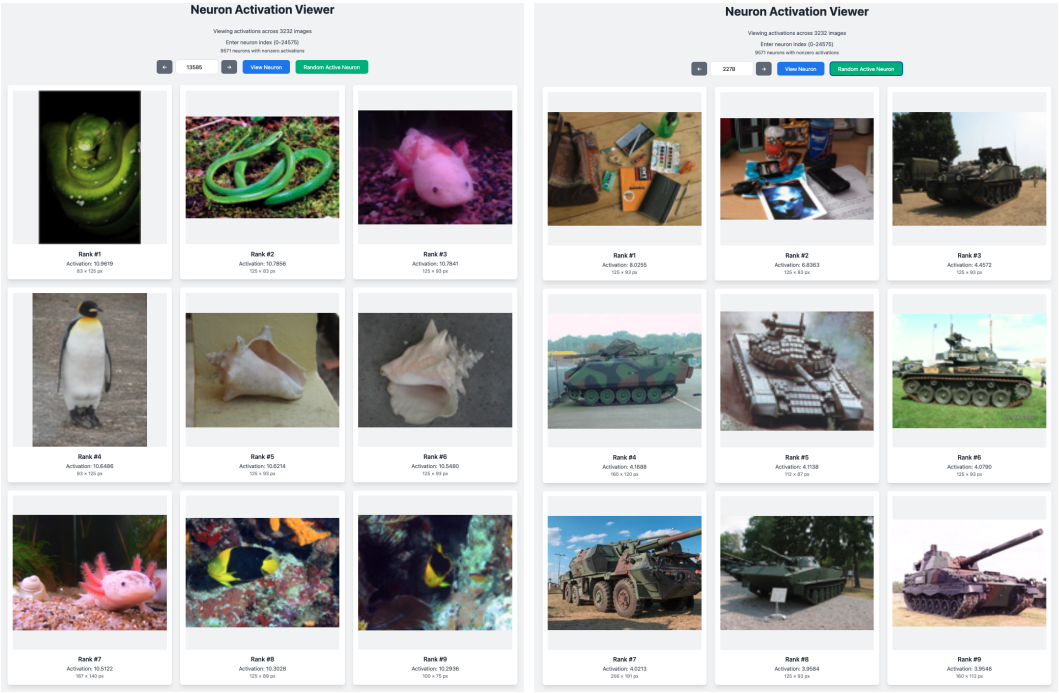

Figure 2: Highest activating images of 2 sample SAE neurons from GPT2 decoder layer 9. Left - Animals, possibly with association to water; Right - Tanks and military vehicles.

4 Evaluation Framework

So far in this paper, we have at least rudimentarily demonstrated full-model multimodal interpretability using SAEs. The next logical stage of finding these interpretable features is to try our same process on models that take both text and image input - such as models built for the Visual Question Answering(VQA) task.

The VQA dataset from Antol et al. (2015) is complete with images, curated questions about the images, and answer evaluation tools. This dataset could be used to evaluate the SAE vectors as-is, but we run into a problem where each very distinct image has a very distinct question. This behavior may be rather unpredictable and hard to generalize within the model.

Here, we propose the idea of a Standard Question Set (SQS): a series of questions that can be asked about *any* image in COCO, ImageNet, or some other image dataset. Just like in many image captioning models, the input text tokens are held constant at ”caption”, allowing us to get interpretability with SAEs, if we control the question similarly, we can isolate the variables and find truly dependent relationships. Alternatively, we could hold the image constant and ask all of the SQS questions. This way, we can form a 2x2 grid of vector activations where each row and column has a unchanging variable in common - which allows us to see what kinds of inputs - questions or images - SAE neurons actually activate for.

For SQS to benefit how we read SAEs, questions would need to be broad, informative, and applicable to almost any image. This often will be open-ended questions - but currently, most such questions are some form of classification question: ”What is the subject of the image?”, ”What is in the background of the image?”, etc. It will be a challenge to create a comprehensive SQS that covers areas across the board (not just classification), but the benefits to multimodal interpretability will be huge.

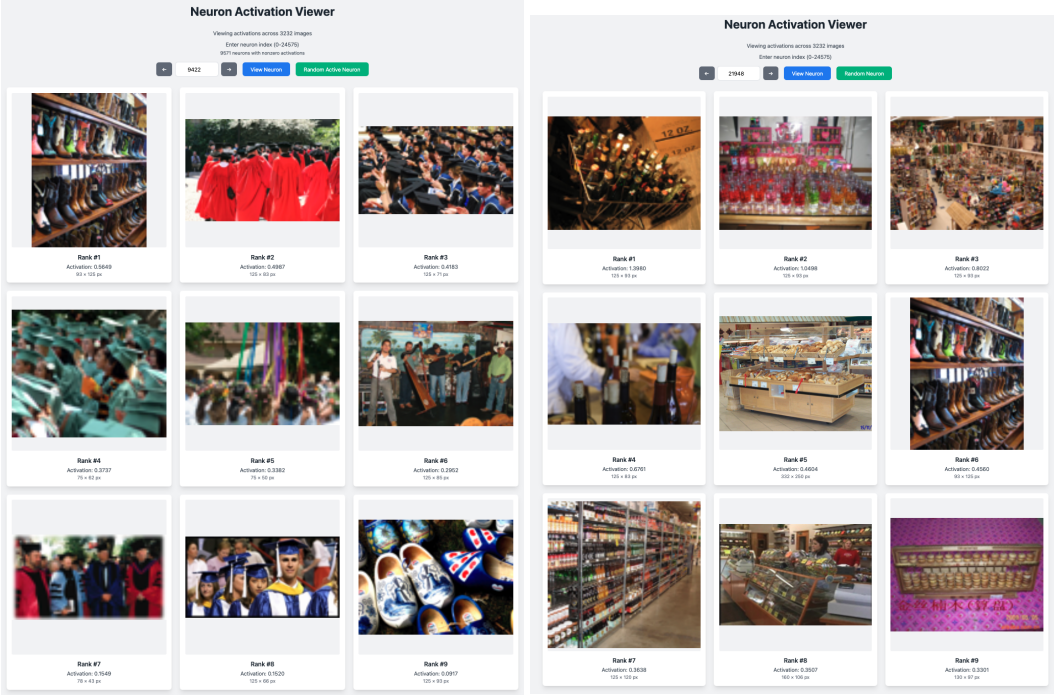

Figure 3: Many neurons represent groups with the similar images, indicating polysemanticity still present.

5 Discussion

This paper is very preliminary work - there are plenty of areas for improvement and further exploration. For example, we chose to use a 500M model for time and cost efficiency. Features would likely be generally more interpretable if the model performed at a higher level. If we wanted to stay on the small side, we could upgrade to a model like the 3B-parameter PaliGemma 2 from Google (see Steiner et al., 2024).

We also trained on a tiny part of the data - about as many training examples as parameters. We would benefit from using 250K-1M+ training examples to properly train the SAE.

Moreover, some additional hyperparameter tuning would be extremely useful. We focused on tuning batch size and learning rate, but the model still showed a lot of polysemanticity in many of the neurons (Figure 3). A higher λ would emphasize the L1 loss, which might encourage sparsity and monosemanticity.

Furthermore, we could explore training SAEs on every single layer's residual stream to see how the input is handled immediately upon reception from the vision encoder and how it transforms through the text model.

If we are to expand the model to take simultaneous text and visual input, we could start with using the VQA dataset verbatim and attempt to manually map prompt-image relationships with neural activations. However the effectiveness of this strategy will probably get capped as there are too many variables - hence if a SQS could be made, it would solve the problem and allow deeper, open understanding.

Using SAEs for interpretability on multi-stage, multimodal models could be extremely vital to the development of more advanced systems - image/text to image/text, models that take video and audio, robotics systems - and furthermore, if all Transformers can be explained, truly open, transparent AI is just around the corner.

References

Ahmed Abdulaal, Hugo Fry, Nina Montaña-Brown, Ayodeji Ijishakin, Jack Gao, Stephanie Hyland, Daniel C. Alexander, and Daniel C. Castro. An x-ray is worth 15 features: Sparse autoencoders for interpretable radiology report generation, 2024. URL https://arxiv.org/abs/2410.03334.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*, 2015.

Joseph Bloom. Saelens. https://github.com/jbloomAus/SAELens, 2024.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023. URL https://arxiv.org/abs/2309.08600.

Lei Gao and Ling Guan. Interpretability of machine learning: Recent advances and future prospects, 2023. URL https://arxiv.org/abs/2305.00537.

Liv Gorton. The missing curve detectors of inceptionv1: Applying sparse autoencoders to inceptionv1 early vision, 2024. URL https://arxiv.org/abs/2406.03662.

hugofry. Towards multimodal interpretability: Learning sparse interpretable features in vision transformers. URL https://www.lesswrong.com/posts/bCtbuWraqYTDtuARg/towards-multimodal-interpretability-learning-sparse-2.

Vandana Rajan, Alessio Brutti, and Andrea Cavallaro. Is cross-attention preferable to self-attention for multi-modal emotion recognition?, 2022. URL https://arxiv.org/abs/2202.09263.

Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery, 2024. URL https://arxiv.org/abs/2407.14499.

Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, Siyang Qin, Reeve Ingle, Emanuele Bugliarello, Sahar Kazemzadeh, Thomas Mesnard, Ibrahim Alabdulmohsin, Lucas Beyer, and Xiaohua Zhai. Paligemma 2: A family of versatile vlms for transfer, 2024. URL https://arxiv.org/abs/2412.03555.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention, 2016. URL https://arxiv.org/abs/1502.03044.

Xin Zhou and Chunyan Miao. Disentangled graph variational auto-encoder for multimodal recommendation with interpretability, 2024. URL https://arxiv.org/abs/2402.16110.

A Appendix

Code used for the project: https://github.com/eric8he/SAE_ViTGPT

Thanks to Joseph Bloom for the SAELens library:
https://github.com/jbloomAus/SAELens

Thanks to Ankur Singh for the base model:
https://huggingface.co/nlpconnect/vit-gpt2-image-captioning