Evaluating Open-Source Sparse Autoencoders on Disentangling Factual Knowledge in GPT-2 Small

Maheep Chaudhary
Pr(Ai)^{2}R Group
maheepchaudhary.research@gmail.com

Atticus Geiger
Pr(Ai)^{2}R Group
atticusg@stanford.edu

Abstract

A popular new method in mechanistic interpretability is to train high-dimensional sparse autoencoders (SAEs) on neuron activations and use SAE features as the atomic units of analysis. However, the body of evidence on whether SAE feature spaces are useful for causal analysis is underdeveloped. In this work, we use the RAVEL benchmark to evaluate whether SAEs trained on hidden representations of GPT-2 small have sets of features that separately mediate knowledge of which country a city is in and which continent it is in. We evaluate four open-source SAEs for GPT-2 small against each other, with neurons serving as a baseline, and linear features learned via distributed alignment search (DAS) serving as a skyline. For each, we learn a binary mask to select features that will be patched to change the country of a city without changing the continent, or vice versa. Our results show that SAEs struggle to reach the neuron baseline, and none come close to the DAS skyline. We release code here: github.com/MaheepChaudhary/SAE_Ravel

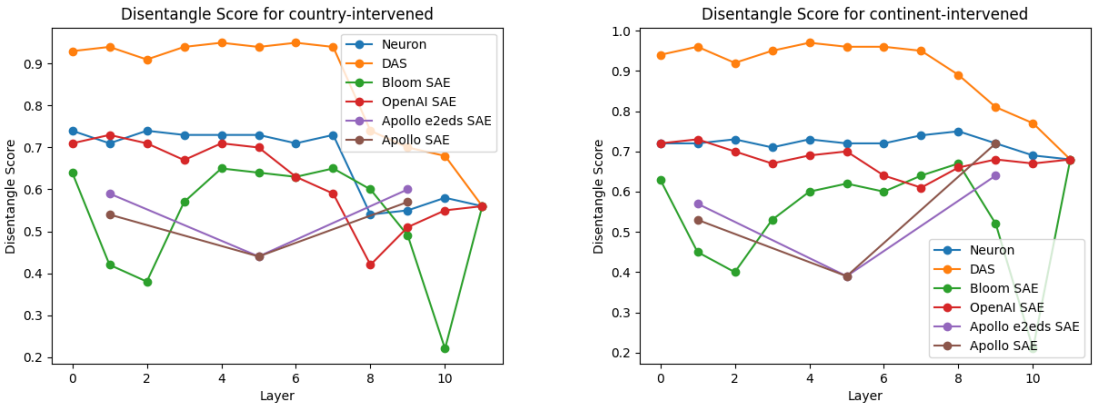
1 Introduction

Individual neurons in neural networks represent many concepts, and individual concepts are represented by many neurons (Gurnee et al., 2023; Bolukbasi et al., 2021; Cammarata et al., 2020; Olah et al., 2020; McClelland et al., 1986a,b; Smolensky, 1988). What, if not neurons, are the relevant meaning-bearing components of neural networks? This is a fundamental question in mechanistic interpretability. A recent, and increasingly popular, unsupervised method for learning features that correspond to intuitive concepts is to train high-dimensional sparse autoencoders (SAEs) on the hidden representations of deep learning models across a wide range of possible inputs (Bricken et al., 2023; Lieberum et al., 2024; Gao et al., 2024). The encoder of an SAE unpacks neurons into a higher dimensional space with sparse linear features that are intended to be better units of analysis.

However, researchers have invested more into scaling SAEs, than evaluating them (Templeton et al., 2024a). In particular, only a handful of works engage with whether SAEs are useful for a causal interpretability analysis (Marks et al., 2024; Engels et al., 2024; Makelov et al., 2024). In this paper, we add to the body of evidence an example of when sparse autoencoders fail to provide a better feature space than neurons. Specifically, we use the RAVEL benchmark (Huang et al., 2024) to evaluate whether the there are sets of SAE features that separately mediate knowledge of which country a city is in and which continent a city is in. We evaluate four publicly available SAEs for GPT-2 small: the Open AI SAE (Gao et al., 2024), two Apollo SAEs (Braun et al., 2024), and the Bloom SAE (Bloom, 2024). As a feature baseline, we use neurons; as a feature skyline, we use linear subspaces trained with distributed alignment search (DAS; Geiger et al. 2024) to disentangle the country knowledge from the continent knowledge.

For each feature space, we train a differentiable binary mask to select features that encode the country of a city, but not the continent, and vice versa. We evaluate the selected features using interchange interventions, where features are fixed to values they would take if a different input were provided. For example, if we fix the 'country' features for the prompt *Toronto is a city in the country* and set them to the value they take for the prompt *Tokyo is a city in*, the output should be *Japan* not *Canada*. If we were to target the 'continent' features instead, the output should be *North America* not *Asia*.

Our results in Figure 1b show that (1) all SAEs struggle to provide a comparable feature space to the neuron baseline, and (3) there is room for improvement, with the DAS skyline outperforming all SAEs.



(a) The disentangle score for 'continent' and 'country' interventions across the layers of GPT-2 small. A disentangle score of 50 is the baseline for an empty intervention. The performance of the DAS skyline goes down after layer 7 because the knowledge about the city is being moved away from the <city> token. The Apollo SAEs are only available for layers 1, 5, and 9.

	Country-Intervened Continent-Preserved						**Continent-Intervened Country-Preserved**					
	Neurons	DAS	Open AI SAE	Bloom SAE	Apollo SAE e2e	Apollo SAE e2e+ds	Neuron	DAS	Open AI SAE	Bloom SAE	Apollo SAE e2e	Apollo SAE e2e+ds
Continent Accuracy	46	93	51	36	24	33	48	94	49	37	24	32
Country Accuracy	96	94	95	49	84	86	97	99	97	52	81	82
Disentangle Score	71	**94**	73	43	54	59	72	**96**	73	45	52	57
Inactive Features	0	0	0.977	0.98	0.966	0.974	0	0	0.977	0.98	0.966	0.974
Non-Intervened Features	0.11	0.24	0.006	0.005	0.01	0.009	0.88	0.79	0.015	0.009	0.023	0.018
Intervened Features	0.89	0.76	0.015	0.009	0.023	0.018	0.12	0.21	0.007	0.005	0.011	0.009
Reconstruction Loss	0	0	152	551	2245	2130	0	0	158	516	2576	2318
Reconstructed Knowledge	100	100	95	56	67	1	100	100	95	47	35	0

(b) GPT-2 small at layer 1. The first three rows are interchange intervention accuracies for RAVEL using learned binary masks to select features. The next three rows are sparsity evaluations that show the proportion of inactive features, intervened on features, and active non-intervened features. The final two rows are reconstruction evaluations that show the models knowledge of cities using a reconstructed representation (no interventions performed) and the average mean-squared error loss after reconstruction.

	Country-Intervened Continent-Preserved						**Continent-Intervened Country-Preserved**					
	Neurons	DAS	Open AI SAE	Bloom SAE	Apollo e2eds SAE	Apollo SAE	Neuron	DAS	Open AI SAE	Bloom SAE	Apollo e2eds SAE	Apollo SAE
Continent Accuracy	49	91	48	53	22	22	46	93	45	48	18	18
Country Accuracy	97	98	79	88	66	66	98	99	79	91	61	61
Disentangle Score	73	**94**	64	70	44	44	72	**96**	62	70	40	39
Inactive Features	0	0	0.951	0.979	0.981	0.98	0	0	0.951	0.979	0.981	0.98
Non-Intervened Features	0.119	0.325	0.017	0.006	0.008	0.009	0.877	0.69	0.031	0.013	0.017	0.017
Intervened Features	0.88	0.674	0.031	0.013	0.017	0.017	0.122	0.309	0.017	0.006	0.008	0.009
Reconstruction Loss	0	0	644	937	2383	2353	0	0	652	1044	2576	2318
Reconstructed Knowledge	100	100	88	77	84	83	100	100	90	86	76	61

(c) GPT-2 small at layer 5. See the caption above from Figure 1b for details on the table structure.

Figure 1: Metrics on the RAVEL test set for interventions performed on the residual stream in GPT-2 small after transformer block above the city token <city>. For each space of features, we learn 'country' features that encode what country a city is in and 'continent' features that encode what continent a city is in. Interventions targeting the 'country' features should change the output for the prompt <city> *is in the country of*, but not <city> *is in the continent of*. Interventions targeting the 'continent' features should do the opposite. The disentangle score is the average of the country and continent accuracies. Neurons serve as a baseline for how easily these two facts are disentangled, and DAS is a supervised feature learning method that serves as a skyline. The SAEs are the methods we seek to evaluate. In sum, using SAE reconstructions harms the knowledge of GPT-2 and SAE features are not better mediators than the baseline of neurons

2 Related Work

Benchmarking SAEs There are many aspects of SAEs to benchmark. To what degree do the features respond precisely and accurately to the natural language labels given to them by auto-interpretability methods (Hernandez et al., 2022; Huang et al., 2023; Schwettmann et al., 2023; Bills et al., 2023; Shaham et al., 2024)? Can we do circuit discovery (Marks et al., 2024; Makelov et al., 2024), representation analysis (Engels et al., 2024), or activation steering (Templeton et al., 2024a) in SAE feature space? Our question is whether SAEs provide a better feature space than neurons for localizing the concepts used by deep learning models.

Interpretability of Knowledge Representations The RAVEL benchmark belongs to a line of research concerned with how factual knowledge is stored within a language model (Geva et al., 2021; Meng et al., 2022; Dai et al., 2022; Meng et al., 2023; Hernandez et al., 2023; Geva et al., 2023). In this paper, we are concerned with how factual knowledge is stored and processed in hidden vector representations during model inference. Activation steering or model editing ask how to control a model, whereas we ask how a model constructs and manipulates representations to control itself.

3 Methodology

3.1 The RAVEL Benchmark

RAVEL (Huang et al., 2024) is an benchmark that evaluates interpretability methods on localizing and disentangling related factual knowledge. We focus on the data for disentangling the country a city is in from the continent it is in.

Filtering Following Huang et al. (2024), we filter out all of the cities that GPT-2 small (Radford et al., 2019) doesn't know both the country and the continent. However, GPT-2 small is not a very capable model, so we give five in context examples when evaluating the knowledge of the model:

"Toronto is a city in the country of Canada. Beijing is a ... `<city>` *is a city in the country of"*

See Appendix A for the full 5-shot prompts. We further filter out multi-token cities to simplify the task and give the SAEs the best chance at success. The resulting dataset contains 40 cities in total.

Interchange Interventions in Feature Space If a set of features contains the knowledge that *Toronto* is in *Canada*, then fixing those features to take on the value they would have for city *Tokyo* make the model think that *Toronto* is in *Japan*. The process of fixing features to take on values they would have for a different input is an *interchange intervention* (Geiger et al., 2020; Vig et al., 2020; Finlayson et al., 2021). Suppose we have base input prompt b and a source input prompt s for a model $\mathcal{M}$ and we want to target features $\mathbf{F}$. Define the interchange intervention as

$$\mathbf{f} = \mathsf{get}(\mathcal{M}, s, \mathbf{F})$$
$$\hat{y} = \mathcal{M}_{\mathbf{F} \leftarrow \mathbf{f}}(b)$$

where $\mathsf{get}(\mathcal{M}, s, \mathbf{F})$ retrieves the value that features $\mathbf{F}$ take on when $\mathcal{M}$ is run with input s and $\mathcal{M}_{\mathbf{F} \leftarrow \mathbf{f}}(b)$ is the output produced when $\mathcal{M}$ is run with input b under intervention $\mathbf{F} \leftarrow \mathbf{f}$.

Counterfactual Labels The label of an interchange intervention example is determined by the concept we think is encoded in the features $\mathbf{F}$ and the mechanism that determines the output given the prompt (Geiger et al., 2021). For our task, the mechanism connecting the knowledge of a city and the expected behavior is simple. If we are intervening on the 'country' features, then the 'country' prompt should have the label from the source y_s and the 'continent' prompt should have the label from the base y_b. If we intervene on the 'continent' feature, we use the opposite labels.

Splits To evaluate a proposed set of 'country' features and 'continent' features, we perform interchange interventions using the RAVEL dataset prompts base and source input prompts. We filtered our dataset down to 40 cities, which can be used to generate 1600 interchange interventions targeting 'country' and 1600 interchange interventions targeting 'continent' (3200 in total). We split the interchange intervention data so that 70% is training, 10% is validation, and 20% is test. Our evaluations are i.i.d. to give SAEs the best chance at success.

3.2 Constructing and Selecting Features

Sparse Autoencoders for Dictionary Learning Sparse autoencoders (SAEs;Bricken et al. 2023; Templeton et al. 2024b) are a unsupervised method for unpacking a hidden vector representation into a higher dimensional, sparsely activated feature space. The hope is that dimensions in this new feature space will correspond to interpretable concepts. SAEs used for this purpose typically have an encoder with a linear transformation followed by a

ReLU and a decoder that is a linear transformation:

$$\bar{x} = x - b_x$$
$$f = \mathsf{ReLU}(W_e\bar{x} + b_e)$$
$$\hat{x} = W_d f + b_d$$

SAEs are optimized jointly to have low reconstruction error and sparse representations:

$$\mathcal{L} = \frac{1}{|X|}\sum_{x\in X} \|x - \hat{x}\|_2^2 + \lambda \|f\|_1$$

Low reconstruction loss ensures that the features faithful to the underlying hidden vector and low sparsity loss is thought to create interpretable features. General purpose SAEs are trained on hidden vector representations created by the model when processing a enormous amount of text data, e.g., an SAE might be trained on residual stream representations created by the second layer of a transformer processing the Pile (Gao et al., 2021).

The Bloom SAE has this standard architecture and training, but the other SAEs are variants. The Open AI SAE is a top-k SAE, which (Gao et al., 2024) show to outperform the standard architecture on the sparsity-reconstruction frontier. A top-k encoder is simply the standard encoder except only the top-k firing features are kept:

$$f = \mathsf{Topk}(\mathsf{ReLU}(W_e\bar{x} + b_e))$$

The two Apollo SAEs have standard architecture, but they are trained with additional loss terms. The Apollo SAE (e2e) is trained with the additional loss objective of the KL-divergence between the output logits of the model before and after reconstruction. The Apollo SAE (e2e + ds) has the logit-based loss in addition to a mean-squared error loss between the residual stream representations in downstream layers before and after reconstruction. (Braun et al., 2024) also report a praeto improvement on the sparsity-reconstruction trade off for end-to-end models.

Distributed Alignment Search SAEs are unsupervised, so features must be further analyzed to determine their conceptual content. In contrast, DAS (Geiger et al., 2024) learns linear features with specific conceptual content via supervision from counterfactual data that describes how a model should act when a concept has been intervened upon.

In particular, DAS learns an orthogonal matrix $\mathbf{R}$ that rotates a hidden vector $\mathbf{h}$, with the dimensions of the rotated space $\mathbf{Rh}$ being the new feature space, i.e. a set of features $\mathbf{F}$ are rows of $\mathbf{Rh}$. We start by randomly initializing $\mathbf{R}$, which renders all features equally meaningless. Then, an interchange intervention is performed on features $\mathbf{F}$ with a base b and source s input prompt pair. Loss is computed from the output of the intervened model:

$$\mathcal{L} = \mathrm{CE}(\mathcal{M}_{\mathbf{F}\leftarrow\mathsf{get}(\mathcal{M},s,\mathbf{F})}(b), y)$$

The expected label y is determined by the concept that we are localizing in $\mathbf{F}$ and the mechanism by which the concept determines behavior. See Section 3.1 for a description of the interchange intervention data. We provide details on hyperparameters in Appendix B.

Differential Binary Masking In order to determine which features $\mathbf{F}$ to select for a given concept ('country' and 'continent' in our case), we use Differential Binary Masking (DBM; Davies et al. 2023; De Cao et al. 2022; Csordás et al. 2021; De Cao et al. 2020) to select features for intervention. Each feature f in the feature space $\mathcal{F}$ is masked with a vector $\mathbf{m}$ which is passed into a sigmoid σ after being scaled by a temperature T:

$$\mathbf{f}_b = \mathsf{get}(\mathcal{M}, \mathbf{F}, b)$$
$$\mathbf{f}_s = \mathsf{get}(\mathcal{M}, \mathbf{F}, s)$$
$$\mathbf{f} = (1 - \sigma(\mathbf{m}/T)) \odot \mathbf{f}_b + \sigma(\mathbf{m}/T) \odot \mathbf{f}_s$$

These masks are trained on an interchange intervention loss objective while the temperature is annealed to make the masks snap to 0 or 1:

$$\mathcal{L} = \mathrm{CE}(\mathcal{M}_{\mathbf{F}\leftarrow\mathbf{f}}(b), y)$$

When we DBM with DAS, the features and the masks are learned simultaneosly.

4 Experiments

Our goal is to find a hidden vector representation in GPT-2 small where the DAS skyline features are significantly better than the neuron baseline, and then evaluate whether SAEs are an improvement on neurons as a unit of analysis. For this reason, we chose to explore the residual stream representations of GPT-2 small above the <city> token in the early layers of the model. We focus our analysis on the residual stream output of layer 1 in GPT-2 small-because the Apollo SAEs are only available for that layer. Results for other layers are similar, and can be found in Appendix **??**. We implement our experiments with nnsight (Fiotto-Kaufman et al., 2024) and pytorch (Paszke et al., 2019).

4.1 Results

In Figure **??** and **??**, we report the interchange intervention accuracy across the layers of GPT-2 small. In Figures 1b and 1c, we present the detailed results for layers 1 and 5 of GPT-2 small. We learned 'country' features and 'continent' features, then we used interchange interventions on those features to evaluate whether they, in fact, store the model's knowledge of the country and continent that a city is in, respectively. When targeting 'country' features for intervention, the 'country' accuracy is high when the intervention changes the output and the 'continent' accuracy is high when the intervention does not change the model output. The opposite is true for interventions on 'continent' features. The 'disentangle score' is the average of the two accuracies. In the middle three rows of the table are sparsity evaluations that report how many features were active and/or intervened upon. In the final two rows of the table are reconstruction evaluations that report the knowledge degradation of GPT-2 small when a reconstructed vector is used and the average reconstruction loss on residual stream vectors above the `<city>` token.

4.2 Discussion

Using representations reconstructed by SAEs degrades the model's knowledge of cities. We can see from the last row in Figures 1b and 1c that using a representation reconstructed by an SAE always degrades the model's knowledge of the countries and continents that cities belong to. For the first layer, we can see that the Bloom SAE and Apollo SAEe2e severely harm the model ($\approx$ -50%) and the Apollo SAE e2e+ds destroys the knowledge entirely. In contrast, the Open AI SAE results in only a small drop in performance (-5%). For the fifth layer, there is less degradation and the Apollo SAE e2e+ds works with Open AI SAE being the best again.

There is a signifigant gap between baseline and skyline; neurons can be improved upon. The skyline provided by DAS at $\approx$95% accuracy for the first 7 layers of GPT-2 small shows that there are separate linear subspaces that encode the country a city is in and the continent a city is in. This means, an SAE with linear features that span these subspaces could achieve performance equivalent to DAS. The neuron baseline at $\approx$70% is significantly worse than the DAS skyline, and shows that there are polysemantic neurons that need to be disentangled by a rotation via an orthogonal matrix.

Current SAEs for GPT-2 small struggle to compete with the neurons. The two Apollo SAEs and Bloom SAE below the neuron baseline across all layers. The 'country' and 'continent' knowledge are even more entangled in the feature spaces provided by these SAEs. The Open AI SAE at $\approx$70% is able to match the performance of the neuron baseline, but not exceed it.

The top-k SAE is the most performant. Our evaluation is limited, however the results do seem to track improvements in SAEs. The Open AI SAE is a top k-SAE, which a performant architecture on sparsity and reconstruction evaluations (Gao et al., 2024). This is in line with our results that the Open AI SAE is the only model that competes with the neuron baseline across all layers. In our limited evaluations, there is no evidence that end-to-end training was helpful for the two Apollo SAEs.

5 Conclusion

We evaluate open-source SAEs on their ability to provide a feature space for GPT-2 hidden representations where knowledge about the country and continent a city is in can be disentangled. We used neurons as a baseline feature space, and a supervised feature learned by DAS as a skyline feature space. While we were able to see meaningful differences in performance between the three SAEs, only one of the evaluated SAEs was able to reach the neuron baseline and none could reach the DAS skyline. We hope this is a small first step in evaluating the usefulness of SAEs for a causal interpretability analysis of deep learning models.

Limitations

In future, we would like to scale the experiments to models with available SAEs including gemma, Mistral, Llama, and Pythia. Furthermore, we hope to use more attributes from the RAVEL dataset, such as *language, gender*, etc. for larger models with more knowledge.

References

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language models can explain neurons in language models.

https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html.

Joseph Bloom. 2024. Open source sparse autoencoders for all residual stream layers of gpt2 small.

Tolga Bolukbasi, Adam Pearce, Ann Yuan, Andy Coenen, Emily Reif, Fernanda Viégas, and Martin Wattenberg. 2021. An interpretability illusion for bert. *Preprint*, arXiv:2104.07143.

Dan Braun, Jordan Taylor, Nicholas Goldowsky-Dill, and Lee Sharkey. 2024. Identifying functionally important features with end-to-end sparse dictionary learning. *Preprint*, arXiv:2405.12241.

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

Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. 2020. Thread: Circuits. *Distill*. Https://distill.pub/2020/circuits.

Róbert Csordás, Sjoerd van Steenkiste, and Jürgen Schmidhuber. 2021. Are neural nets modular? inspecting functional modularity through differentiable weight masks. In *International Conference on Learning Representations*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *acl*.

Xander Davies, Max Nadeau, Nikhil Prakash, Tamar Rott Shaham, and David Bau. 2023. Discovering variable binding circuitry with desiderata. *Preprint*, arXiv:2307.03637.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. 2020. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online. Association for Computational Linguistics.

Nicola De Cao, Leon Schmid, Dieuwke Hupkes, and Ivan Titov. 2022. Sparse interventions in language models with differentiable masking. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 16–27, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Joshua Engels, Isaac Liao, Eric J. Michaud, Wes Gurnee, and Max Tegmark. 2024. Not all language model features are linear. *CoRR*, abs/2405.14860.

Matthew Finlayson, Aaron Mueller, Sebastian Gehrmann, Stuart Shieber, Tal Linzen, and Yonatan Belinkov. 2021. Causal analysis of syntactic agreement mechanisms in neural language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1828–1843, Online. Association for Computational Linguistics.

Jaden Fiotto-Kaufman, Alexander R Loftus, Eric Todd, Jannik Brinkmann, Caden Juang, Koyena Pal, Can Rager, Aaron Mueller, Samuel Marks, Arnab Sen Sharma, Francesca Lucchetti, Michael Ripa, Adam Belfki, Nikhil Prakash, Sumeet Multani, Carla Brodley, Arjun Guha, Jonathan Bell, Byron Wallace, and David Bau. 2024. Nnsight and ndif: Democratizing access to foundation model internals. *Preprint*, arXiv:2407.14561.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

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. *CoRR*, abs/2406.04093.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 9574–9586.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah D. Goodman. 2024. Finding alignments between interpretable causal variables and distributed neural representations. *Preprint*, arXiv:2303.02536.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *Preprint*, arXiv:2304.14767.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *emnlp*.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *Preprint*, arXiv:2305.01610.

Evan Hernandez, Belinda Z Li, and Jacob Andreas. 2023. Measuring and manipulating knowledge representations in language models. *arXiv preprint arXiv:2304.00740*.

Evan Hernandez, Sarah Schwettmann, David Bau, Teona Bagashvili, Antonio Torralba, and Jacob Andreas. 2022. Natural language descriptions of deep visual features. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jing Huang, Atticus Geiger, Karel D'Oosterlinck, Zhengxuan Wu, and Christopher Potts. 2023. Rigorously assessing natural language explanations of neurons. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@EMNLP 2023, Singapore, December 7, 2023*, pages 317–331. Association for Computational Linguistics.

Jing Huang, Zhengxuan Wu, Christopher Potts, Mor Geva, and Atticus Geiger. 2024. Ravel: Evaluating interpretability methods on disentangling language model representations. *Preprint*, arXiv:2402.17700.

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. *Preprint*, arXiv:2408.05147.

Aleksandar Makelov, George Lange, and Neel Nanda. 2024. Towards principled evaluations of sparse autoencoders for interpretability and control. *Preprint*, arXiv:2405.08366.

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. *CoRR*, abs/2403.19647.

James L. McClelland, David E. Rumelhart, and PDP Research Group. 1986a. *Parallel Distributed Processing, Volume 2: Explorations in the Microstructure of Cognition: Psychological and Biological Models*. The MIT Press.

James L. McClelland, David E. Rumelhart, and PDP Research Group. 1986b. *Parallel Distributed Processing, Volume 2: Explorations in the Microstructure of Cognition: Psychological and Biological Models*. The MIT Press.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. 2020. Zoom in: An introduction to circuits. *Distill*. Https://distill.pub/2020/circuits/zoom-in.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Sarah Schwettmann, Tamar Rott Shaham, Joanna Materzynska, Neil Chowdhury, Shuang Li, Jacob Andreas, David Bau, and Antonio Torralba. 2023. FIND: A function description benchmark for evaluating interpretability methods. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Tamar Rott Shaham, Sarah Schwettmann, Franklin Wang, Achyuta Rajaram, Evan Hernandez, Jacob Andreas, and Antonio Torralba. 2024. A multimodal automated interpretability agent. *CoRR*, abs/2404.14394.

Paul Smolensky. 1988. On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1):1–23.

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, Shan Carter, Chris Olah, and Tom Henighan. 2024a. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*.

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,

Shan Carter, Chris Olah, and Tom Henighan. 2024b. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

A Evaluation Details

To enhance, the prediction capability of GPT-2 using in-context learning, we use 5-shot prompt for both the attributes. Specifically, for country attribute, we prepare a template as: *"Toronto is a city in the country of Canada. Beijing is a city in the country of China. Miami is a city in the country of the United States. Santiago is a city in the country of Chile. London is a city in the country of England.* `<city>` *is a city in the country of"*.

Similarly, to support the prediction of continent, we also prepare a similar template for the model as: *"Toronto is a city in the continent of North America. Beijing is a city in the continent of Asia. Miami is a city in the continent of North America. Santiago is a city in the continent of South America. London is a city in the continent of Europe.* `<city>` *is a city in the continent of"*. The `<city>` is replaced with the city name in the dataset to make several samples to make the data for both the country and continent attributes.

Eventually, we prepare the final dataset consisting of base and source sentences, with their corresponding labels to evaluate different techniques. In each example, either the 'country' is targeted for intervention or the 'continent' is. When a prompt is for targeted attribute, the intervention should change the output to match the source city. When the prompt is for the other attribute, the intervention should not change the output.

B Hyperparameters and Compute

We used these parameters for DBM and DBM+DAS training. Batch size of 16. Temperature is annealed linearly from 10 to 0.1. Training was for 20 epochs. Learning rate is 0.001.

A masking experiment takes 1 hour approx to run. Three layers had 4 experiments with a run for for each intervention so 4*2 experiments. Layer 1 had a total of 6 experiments with two interventions each. Total time: 1x3x4x2 + 1x6x2 = 36 hours on a 24GB Nvidia RTX A5000

C Full Reconstruction Evaluation

Layers	Bloom SAE Country	Bloom SAE Continent	OpenAI SAE Country	OpenAI SAE Continent	Apollo SAE Country	Apollo SAE Continent	Apollo SAE e2e+ds Country	Apollo SAE e2e+ds Continent
Layer 0	400.87	413.03	102.91	104.2	-	-	-	-
Layer 1	551.28	516.5	151.83	158.03	2245.57	2307.15	2129.71	2123.09
Layer 2	698.25	681.64	217.13	219.78	-	-	-	-
Layer 3	876.36	814.99	330.43	336.34	-	-	-	-
Layer 4	890.41	869.71	449.33	458.82	-	-	-	-
Layer 5	936.77	1044.33	643.82	651.67	2383.14	2576.08	2353.61	2318.49
Layer 6	1178.01	1531.46	839.68	837.81	-	-	-	-
Layer 7	4640.78	7757.06	1218.99	1211.81	-	-	-	-
Layer 8	19556.78	26810.38	1727.77	1723.93	-	-	-	-
Layer 9	27877.84	36537.93	2304.84	2311.26	5276.6	6038.87	2569.59	2665.5
Layer 10	532812.74	571233.39	3296.77	3467.73	-	-	-	-
Layer 11	846887.04	859555.3	4833.99	4893.55	-	-	-	-

Table 1: The table above denotes the reconstruction loss for country and continent dataset separately for each SAE.

Layers	Bloom SAE Country	Bloom SAE Continent	OpenAI SAE Country	OpenAI SAE Continent	Apollo SAE Country	Apollo SAE Continent	Apollo SAE e2e+ds Country	Apollo SAE e2e+ds Continent
Layer 0	0.9375	0.890625	0.9642857142857143	0.9609375	-	-	-	-
Layer 1	0.5625	0.46875	0.9464285714285714	0.9453125	0.6696428571428571	0.3515625	0.008928571428571428	0.0
Layer 2	0.5267857142857143	0.5390625	0.9553571428571429	0.9140625	-	-	-	-
Layer 3	0.7142857142857143	0.78125	0.9196428571428571	0.890625	-	-	-	-
Layer 4	0.7946428571428571	0.8984375	0.9196428571428571	0.875	-	-	-	-
Layer 5	0.7678571428571429	0.859375	0.875	0.8984375	0.8392857142857143	0.7578125	0.8303571428571429	0.609375
Layer 6	0.7946428571428571	0.78125	0.8125	0.7421875	-	-	-	-
Layer 7	0.875	0.765625	0.7857142857142857	0.703125	-	-	-	-
Layer 8	0.8571428571428571	0.7578125	0.8571428571428571	0.8984375	-	-	-	-
Layer 9	0.6696428571428571	0.5078125	0.9107142857142857	0.9609375	0.9464285714285714	0.9453125	0.875	0.90625
Layer 10	0.23214285714285715	0.03125	0.9732142857142857	0.9765625	-	-	-	-
Layer 11	1.0	1.0	1.0	1.0	-	-	-	-

Table 2: The table above denotes the accuracy for country and continent dataset after intervention for each SAE