

5 DISCUSSION

We considered the activation vectors from different layers as different training examples, so we passed $n_L n_T$ vectors of length d to the autoencoder, where n_T is the number of tokens, n_L is the number of layers, and d is the dimension of the residual stream. This approach might be called a ‘data-stacked’ MLSAE. An alternative would be a ‘feature-stacked’ MLSAE, i.e., to concatenate the activation vectors from different layers into a single vector of dimension $n_L d$. This alternative might be better suited to capturing the notion of ‘cross-layer superposition,’ which we take to mean a small number of simultaneously active sparse features at multiple layers encoding a single meaningful concept (Olah, 2024; Templeton et al., 2024).

We began by pursuing the feature-stacked approach but discarded it. The essential issue is that a single set of sparse features describes the residual stream activations at every layer, which makes it difficult to understand how information flows through a transformer. For example, it would not be possible to plot the activations of sparse features across layers. Moreover, to compute this set of features, one must first compute the activations at every layer, which makes it more difficult to evaluate performance by traditional measures like single-layer reconstruction errors. Finally, the information encoded at one token position may differ substantially between layers due to self-attention. In the early layers, the representation is likely to primarily encode the input token and position embedding, whereas in the later layers, the representation may encode more complex properties of the surrounding context. It is not immediately apparent that jointly encoding this information by a single SAE is sensible. Instead, one might wish to separately capture the different information present at a token position across layers, which is allowed with our data-stacked approach.

6 CONCLUSION

We introduced the multi-layer SAE (MLSAE), where we train a single SAE on the activations at every layer of the residual stream. This allowed us to study both how information is represented within a single transformer layer and how information flows through the residual stream.

We confirmed that residual stream activations are relatively similar across layers by looking at cosine similarities before considering the distributions of latent activations over layers. When aggregating over a large sample of ten million tokens, we observed that most latents were active at multiple layers, but for a single prompt, most latent activations were isolated to a single layer. To quantify these observations, we computed the fraction of the total variance explained by individual latents and the fraction of the variance for an individual latent explained by individual tokens. This analysis confirmed that the degree to which latents are active at multiple layers when aggregating over tokens was large, increasing with the model size and expansion factor, and that the fraction of the variance explained by individual tokens was small.

Understanding how representations change as they flow through transformers is critical to identifying meaningful circuits, which is a core task of mechanistic interpretability. Despite the utility of the residual stream perspective, our results demonstrate that representation drift, and perhaps the increasing magnitude of changes to the residual stream across layers, is a significant obstacle to identifying meaningful computational variables with SAEs. Nevertheless, we argue that an approach such as the MLSAE, which considers the representations at multiple layers in parallel, is necessary for future methods that seek to interpret the internal computations of transformer language models.

7 ACKNOWLEDGEMENTS

Tim Lawson and Lucy Farnik were supported by the UKRI Centre for Doctoral Training in Interactive Artificial Intelligence (EP/S022937/1). This work was carried out with HPC systems provided by the Advanced Computing Research Centre at the University of Bristol. We also thank Dr. Stewart, whose philanthropy supported the compute resources used.

8 REPRODUCIBILITY STATEMENT

We release our code to train and analyze MLSAEs at <https://github.com/tim-lawson/mlsae>, and the models described in the paper at <https://huggingface.co/papers/2409.04185>. Section 3 and Appendix A describe the training setup, Section 4.1 and Appendix B describe the evaluation metrics, and Section 4.3 and Appendix C describe our analyses.

REFERENCES

- Anthony J. Bell and Terrence J. Sejnowski. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 7(6):1129–1159, November 1995. ISSN 0899-7667. doi: 10.1162/neco.1995.7.6.1129. URL <https://ieeexplore.ieee.org/abstract/document/6796129>. Conference Name: Neural Computation.
- Anthony J. Bell and Terrence J. Sejnowski. The “independent components” of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, December 1997. ISSN 0042-6989. doi: 10.1016/S0042-6989(97)00121-1. URL <https://www.sciencedirect.com/science/article/pii/S0042698997001211>.
- Nora Belrose. EleutherAI/sae, May 2024. URL <https://github.com/EleutherAI/sae>.
- Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting Latent Predictions from Transformers with the Tuned Lens, November 2023. URL <http://arxiv.org/abs/2303.08112>. arXiv:2303.08112 [cs].
- Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usman Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 2397–2430. PMLR, July 2023. URL <https://proceedings.mlr.press/v202/biderman23a.html>. ISSN: 2640-3498.
- Dan Braun, Jordan Taylor, Nicholas Goldowsky-Dill, and Lee Sharkey. Identifying Functionally Important Features with End-to-End Sparse Dictionary Learning, May 2024. URL <http://arxiv.org/abs/2405.12241>. arXiv:2405.12241 [cs].
- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, and Amanda Askell. Towards Monosemanticity: Decomposing Language Models With Dictionary Learning, 2023. URL <https://transformer-circuits.pub/2023/monosemantic-features>.
- Maheep Chaudhary and Atticus Geiger. Evaluating Open-Source Sparse Autoencoders on Disentangling Factual Knowledge in GPT-2 Small, September 2024. URL <http://arxiv.org/abs/2409.04478>. arXiv:2409.04478 [cs].
- Arthur Conny, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards Automated Circuit Discovery for Mechanistic Interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, December 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/34e1dbe95d34d7ebaf99b9bcab5b2be-Abstract-Conference.html.
- Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse Autoencoders Find Highly Interpretable Features in Language Models, October 2023. URL <http://arxiv.org/abs/2309.08600>. arXiv:2309.08600 [cs].
- Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders Find Interpretable LLM Feature Circuits, June 2024. URL <http://arxiv.org/abs/2406.11944>. arXiv:2406.11944 [cs].
- Nelson Elhage, Neel Nanda, Catherine Olsson, T. Henighan, N. Joseph, B. Mann, A. Askell, Y. Bai, A. Chen, and T. Conerly. A Mathematical Framework for Transformer Circuits, 2021. URL <https://transformer-circuits.pub/2021/framework/index.html>.

- 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. Toy Models of Superposition, September 2022. URL <http://arxiv.org/abs/2209.10652>. arXiv:2209.10652 [cs].
- Joshua Engels, Isaac Liao, Eric J. Michaud, Wes Gurnee, and Max Tegmark. Not All Language Model Features Are Linear, May 2024. URL <http://arxiv.org/abs/2405.14860>. arXiv:2405.14860 [cs].
- Javier Ferrando, Gabriele Sarti, Arianna Bisazza, and Marta R. Costa-jussà. A Primer on the Inner Workings of Transformer-based Language Models, May 2024. URL <http://arxiv.org/abs/2405.00208>. arXiv:2405.00208 [cs].
- 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. The Pile: An 800GB Dataset of Diverse Text for Language Modeling, December 2020. URL <http://arxiv.org/abs/2101.00027>. arXiv:2101.00027 [cs].
- Leo Gao, Tom Dupré la Tour, and Jeffrey Wu. openai/sparse_autoencoder, December 2023. URL https://github.com/openai/sparse_autoencoder.
- Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders, June 2024. URL <http://arxiv.org/abs/2406.04093>. arXiv:2406.04093 [cs].
- Jorge García-Carrasco, Alejandro Maté, and Juan Carlos Trujillo. How does GPT-2 Predict Acronyms? Extracting and Understanding a Circuit via Mechanistic Interpretability. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, pp. 3322–3330. PMLR, April 2024. URL <https://proceedings.mlr.press/v238/garcia-carrasco24a.html>. ISSN: 2640-3498.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The Llama 3 Herd of Models, November 2024. URL <http://arxiv.org/abs/2407.21783>.
- Zhengfu He, Xuyang Ge, Qiong Tang, Tianxiang Sun, Qinyuan Cheng, and Xipeng Qiu. Dictionary Learning Improves Patch-Free Circuit Discovery in Mechanistic Interpretability: A Case Study on Othello-GPT, February 2024. URL <http://arxiv.org/abs/2402.12201>. arXiv:2402.12201 [cs].
- Stefan Heimersheim and Alex Turner. Residual stream norms grow exponentially over the forward pass, May 2023. URL <https://www.alignmentforum.org/posts/8mizBCm3dyc432nK8/residual-stream-norms-grow-exponentially-over-the-forward>.
- Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of Relation Decoding in Transformer Language Models, February 2024. URL <http://arxiv.org/abs/2308.09124>. arXiv:2308.09124 [cs].
- Jing Huang, Zhengxuan Wu, Christopher Potts, Mor Geva, and Atticus Geiger. RAVEL: Evaluating Interpretability Methods on Disentangling Language Model Representations, August 2024. URL <http://arxiv.org/abs/2402.17700>. arXiv:2402.17700 [cs].
- A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411–430, June 2000. ISSN 0893-6080. doi: 10.1016/S0893-6080(00)00026-5. URL <https://www.sciencedirect.com/science/article/pii/S0893608000000265>.
- Stanisław Jastrzębski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual Connections Encourage Iterative Inference, March 2018. URL <http://arxiv.org/abs/1710.04773>. arXiv:1710.04773 [cs].
- Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017. URL <http://arxiv.org/abs/1412.6980>. arXiv:1412.6980 [cs].