STEERING FINE-TUNING GENERALIZATION WITH TARGETED CONCEPT ABLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

During fine-tuning, multiple solutions may emerge which perform similarly on training data but generalize differently out of distribution. For instance, a deceptive model may be indistinguishable from an aligned model during training, but perform catastrophically at deployment. We present a novel technique for controlling what models learn during fine-tuning by identifying and ablating specific sparse autoencoder latents that represent undesired concepts. Our approach steers models toward intended generalizations when multiple policies correctly fit the training data. We evaluate our method on two tasks, significantly outperforming baselines: a gender bias task containing spurious correlations and a double multiple choice task where models must learn to focus on intended questions while ignoring others. On gender bias, our method completely eliminates spurious correlations, leading to strong performance out of distribution. In double multiple choice, it succeeds in 12 out of 16 scenarios. Our results mark an initial step toward using interpretability techniques to ensure the safe and reliable deployment of frontier AI systems.

1 INTRODUCTION

Models often learn undesired behaviors during fine-tuning. For example, training AI assistants with human feedback can encourage them to match user beliefs instead of giving truthful answers (Sharma et al., 2023). One way to prevent models from learning undesired behaviors is to remove the data responsible for them, and there is a large body of research aimed at localizing subsets of training data responsible for a given model behavior (Grosse et al., 2023; Park et al., 2023; Ilyas et al., 2022). However, it is possible that structural factors lead to intended and unintended behaviors being deeply linked across an entire training corpus, which would make it impossible to remove unintended behaviors by removing corresponding training data. For example, data for different classes might come from different distributions (Zech et al., 2018). As AI systems become more powerful, controlling how a model generalizes from training data will become an increasingly important problem (Burns et al., 2023; Hase et al., 2024).

In this work we present a method that uses interpretability techniques to control what a model learns during fine-tuning. We address the case where there are multiple policies that are correct on *all* training samples but have extremely different generalizations. To do so, we decompose model activations into interpretable directions using sparse autoencoders (SAEs) (Bricken et al., 2023; Cunningham et al., 2023). We identify unwanted concepts and ablate them during fine-tuning. This steers the model towards the intended solution.

We evaluate our method on two types of multiple choice tasks. The first task involves pronoun completion using data that contains a spurious correlation between occupation and gender. The second is a double multiple choice task where each prompt contains two questions on different topics, and the model must learn to focus on one intended question while ignoring the other. We successfully use our method to train LLMs that generalize correctly in 13 out of 17 scenarios. Our results demonstrate that by identifying and ablating specific SAE latents during fine-tuning, we can effectively prevent models from learning unintended generalizations from the training data while preserving their ability to learn the intended task.

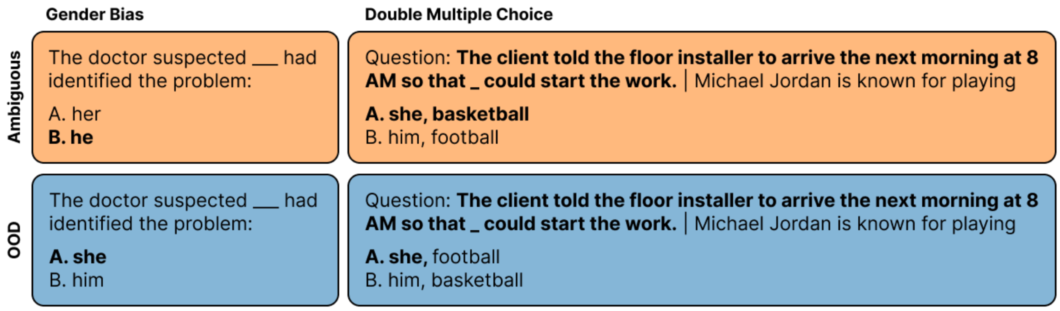

Figure 1: Example inputs from our multiple choice datasets. D_{amb} contains a spurious correlation that allows the model to learn either the intended task or an unintended shortcut. The test dataset D_{OOD} breaks this correlation to evaluate whether the model learned the correct generalization.

2 Background and Related Work

Sparse autoencoders (SAEs). Recent work in interpretability employs techniques from sparse dictionary learning (Olshausen & Field, 1997; Lee et al., 2006) to decompose language model activations into a set of latent vectors (Cunningham et al., 2023; Bricken et al., 2023). While recent work has improved upon initial SAE baselines (Gao et al., 2024; Rajamanoharan et al., 2024), SAEs have shown limited practical improvements outside of narrow interpretability tasks (Wu et al., 2025; Farrell et al., 2024; Menon et al., 2025; Marks et al., 2024).

Removing unintended correlations or concepts. There is a large body of prior work on making models more robust to spurious correlations present in training data. Many such techniques require: access to an additional set of labels to distinguish the intended from unintended generalizations (Nam et al., 2020; 2022; Sagawa et al., 2020), the spurious correlation to only be present in some of the data (Yaghoobzadeh et al., 2021; Utama et al., 2020), or an additional classifier for the unintended label (Kim et al., 2019). Prior work on unlearning also assumes access to supervised data that isolates an unlearning target (Belrose et al., 2023; Guo et al., 2024; Iskander et al., 2023; Wang et al., 2020; Ravfogel et al., 2020; 2022; Thaker et al., 2024). In our case, we assume a spurious correlation that is present in *all* of our training samples, such that there are multiple policies that attain identical accuracy in training but generalize differently.

3 Formulation

Our problem assumes that we have a labeled ambiguous dataset $D_{\text{amb}} = \{(x, y)\}$ such that there are multiple ways to predict the label y from the input x. For simplicity, we consider cases where, due to a spurious correlation present in all of the training data, there are two possible generalizations, an *intended* generalization and an *unintended* generalization. Our goal is to train a model to predict the output in the intended way by only fine-tuning it on D_{amb}. To test the model's generalization, we create a dataset D_{OOD} (out of distribution) where only the intended generalization results in high accuracy, while the unintended generalization results in low accuracy. To validate in-distribution performance, we also use a dataset D_{val} of the same form as D_{amb}. We use two types of multiple choice tasks to test our method, with D_{amb} and D_{OOD} examples shown in Figure 1.

Gender bias is a multiple choice task in which the model selects between two gendered pronouns to complete a sentence. The dataset has a correlation between the subject's gender and the grammatically correct answer. In D_{amb} and D_{val}, the correct pronoun is always male for a doctor and female for a nurse. D_{OOD} has the inverted gender correlation; to correctly generalize, the model should learn to select the grammatically correct pronoun regardless of the subject. The dataset prompts were generated using Claude 3.5 Sonnet, inspired by Perez et al. 2022 and De-Arteaga et al. 2019.

Double multiple choice consists of multiple choice problems where each question is composed of two sub-questions from different datasets. We use four different datasets for the individual questions. We formalize the task using tuples (Q_a, Q_b, Q^*), where Q_a is the first question, Q_b is the second question and $Q^* \in \{Q_a, Q_b\}$ is the intended question. This results in 24 possible (Q_a, Q_b, Q^*)

combinations (we exclude those where $Q_a = Q_b$). Answers are comma separated combinations of answers to (Q_a, Q_b). In D_{amb} and D_{val}, the correct answers to both questions are in the same selection. In D_{OOD}, the options each contain one correct and one incorrect answer. We filter out the (Q_a, Q_b, Q^*) combinations that achieve higher than 90% accuracy when trained without interventions, leaving 16 combinations. Although the task is unnatural, it provides a simple method for creating many unintended correlations to test our method.

4 METHODS

Given D_{amb}, we use sparse autoencoders to identify causally relevant latents for predicting the correct answer. We interpret the latents and identify ones related to the unintended generalization, then fine-tune the model directly on D_{amb} while ablating these latents at each forward pass. Specifically:

1. **Find causally important latents** by attribution effects over the whole dataset D_{amb}. We calculate attribution scores by approximating the effect that ablating each latent would have on an output metric m as in Marks et al. (2024), which applies attribution patching (Nanda, 2023; Syed et al., 2023)) to SAE latents:

$$E = m\left(x^*|\text{do}(z=0)\right) - m(x^*) \approx \sum_t \nabla_z m|_{z_t = z_t^*} \cdot z_t^*, \tag{1}$$

   where z is the SAE latent activation, x is the model input, and x^* and z^* denote values under a given input. $m\left(x^*|\text{do}(z=0)\right)$ refers to the value m takes under input x^* when we intervene in the forward pass setting $z = 0$. The subscript t refers to the token position of the activations. In our case, the metric m is the logit difference between the correct answer token and the incorrect answer token (usually '`_A`' or '`_B`'). We average effects over D_{amb} inputs to estimate the expected value over the dataset.

2. **Interpret and select latents** by inspecting top activating examples. We select the top 100 latents by attribution effect, then filter for relevance on the unintended generalization task. We automatically interpret top latents with Llama 3.3 70B (Grattafiori et al., 2024) using a modified pipeline from Paulo et al. (2024). For each task, we query for relevant explanations using a text embedding model, then further filter for explanations with high interpretability scores. We use simulation scoring from Bills et al. (2023), with Qwen 2.5 7B as our simulator (Qwen et al., 2025). As a baseline, we also manually interpret the same sets of latents using activating examples from Neuronpedia (Lin, 2023). See Appendix B for further implementation details.

3. **Ablate unintended latents while fine-tuning** on D_{amb}. At each forward pass, we use the SAE to encode model activations and obtain SAE latents. We set the unintended activations to zero, then use the decoder to obtain new model activations. We add the SAE reconstruction error to the model activations.

At runtime, it is an empirical question as to whether we should continue to ablate unintended latents as we do during fine-tuning or if ablation during fine-tuning is sufficient for the model to learn the intended task. This simplifies inference since there is no need to ablate latents after training. As a baseline, we compare against random ablations, where we ablate an equal number of latents at random from the latents with top 100 effects. We also compare against fine-tuning the model without interventions and only ablating the latents during test time.

5 RESULTS

We conduct our experiments on Gemma 2 2B (Team et al., 2024) using a suite of residual stream SAEs from Gemma Scope (Lieberum et al., 2024). Results are averaged over 5 different seeds and error bars show standard error of the mean unless noted otherwise.

Baseline performance. On gender bias, Gemma achieves perfect validation accuracy but learns to make gendered completions, achieving just 8% accuracy on D_{OOD}. Across all double multiple choice combinations, Gemma gets at least 97% validation accuracy. We filter for task combinations where the model achieves less than 90% accuracy on the intended question on D_{OOD}. Figure 2

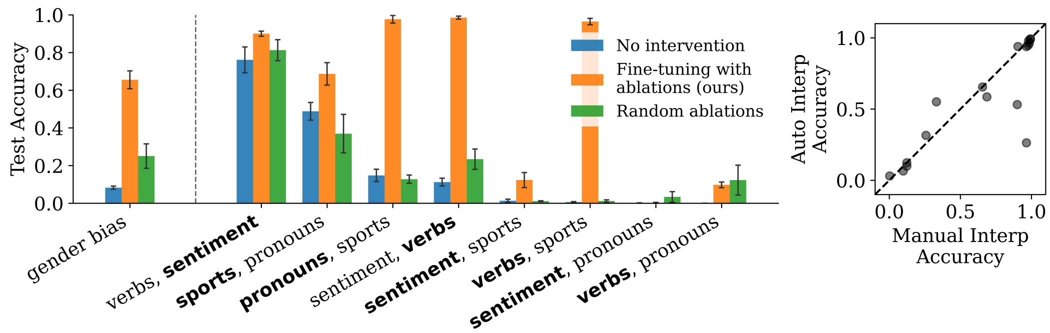

Figure 2: (Left) Accuracy on D_{OOD} for the gender bias and multiple choice tasks. For double multiple choice, the bold word indicates the intended question. (Right) Accuracy on D_{OOD} across all tasks, comparing ablations using manually and automatically interpreted latents. (Both) Accuracy is tested using the model that was fine-tuned with ablations, without ablating during testing.

shows eight of the sixteen tasks; all pairs are represented, and we choose the pair ordering that had lower no-intervention accuracy. See Appendix E for more.

Interpreting latents with highest attribution. When interpreting the top 100 latents, we find many that appear important for answering multiple choice questions; for example, features that detect or promote '`A`', '`B`', and other similar tokens. We also find latents relevant to intended and unintended task features. For the gender bias task, we identify 6 unintended latents, mostly activating on female or nurse related words. For the double multiple choice task, we find 2-27 latents depending on the data set and question order. Results are similar for automated interpretations. See Appendix A for a detailed breakdown.

Training with unintended features ablated. On gender bias, the model trained with ablations learns the intended generalization, achieving 99.1% accuracy on D_{val} and 86.4% accuracy on D_{OOD}. For double multiple choice, out of the 16 task combinations we found improvement in intended question accuracy D_{OOD} in 12 cases. Figure 2 shows the results for the gender bias task and double multiple choice question tasks. Full results are shown in Appendix D and E. In four of the cases, the ablation did not work; these correspond to ablating the pronoun latents when the intended question is sentiment or verbs. Figure 2 shows that ablating automatically interpreted latents yields similar accuracies.

Baselines. Random ablations prove ineffective (Figure 2). Another way to alter the unintended generalization is to ablate features at evaluation on a model fine-tuned without ablations. This performs worse than intervening during training across our tasks (Appendix E). It also requires constant modification of the model at inference which is impractical for efficient and reliable deployment.

6 Conclusion

We demonstrate a method for guiding a language model's generalization by ablating certain subspaces during training. The approach performs strongly on toy tasks, but it faces certain limitations in scaling to larger, complex scenarios. Our work is an initial step in the direction of using interpretability methods for building trust into language models. By controlling generalization from training data, we provide more robust guarantees for safety and reliability in the real world.

7 Limitations

Locating full concept subspaces for ablation is challenging due to limitations of SAEs. Engels et al. (2024) find SAE error is pathological and Menon et al. (2025) show that SAEs reflect inductive biases of their pipeline, not true features of model computation. Additionally, automated interpretability pipelines fail to capture functional features whose explanations aren't obvious from top activations.

REFERENCES

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. Leace: Perfect linear concept erasure in closed form. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 66044–66063. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/d066d21c619d0a78c5b557fa3291a8f4-Paper-Conference.pdf.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. *URL https://openaipublic. blob. core. windows. net/neuron-explainer/paper/index. html.(Date accessed: 14.05. 2023)*, 2, 2023.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Chris Olah. Towards monosemanticity: Decomposing language models with dictionary learning, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeff Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision, 2023. URL https://arxiv.org/abs/2312.09390.

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.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 120–128, 2019.

Joshua Engels, Logan Riggs, and Max Tegmark. Decomposing the dark matter of sparse autoencoders, 2024. URL https://arxiv.org/abs/2410.14670.

Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. Applying sparse autoencoders to unlearn knowledge in language models. *arXiv preprint arXiv:2410.19278*, 2024.

Jaden Fiotto-Kaufman, Alexander R Loftus, Eric Todd, Jannik Brinkmann, Caden Juang, Koyena Pal, Can Rager, Aaron Mueller, Samuel Marks, Arnab Sen Sharma, et al. Nnsight and ndif: Democratizing access to foundation model internals. *arXiv preprint arXiv:2407.14561*, 2024.

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. *arXiv preprint arXiv:2406.04093*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra,

Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan

Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, Evan Hubinger, Kamilė Lukošiūtė, Karina Nguyen, Nicholas Joseph, Sam McCandlish, Jared Kaplan, and Samuel R. Bowman. Studying large language model generalization with influence functions, 2023. URL `https://arxiv.org/abs/2308.03296`.

Phillip Guo, Aaquib Syed, Abhay Sheshadri, Aidan Ewart, and Gintare Karolina Dziugaite. Mechanistic unlearning: Robust knowledge unlearning and editing via mechanistic localization, 2024. URL `https://arxiv.org/abs/2410.12949`.

Peter Hase, Mohit Bansal, Peter Clark, and Sarah Wiegreffe. The unreasonable effectiveness of easy training data for hard tasks, 2024. URL `https://arxiv.org/abs/2401.06751`.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting predictions from training data, 2022. URL `https://arxiv.org/abs/2202.00622`.

Shadi Iskander, Kira Radinsky, and Yonatan Belinkov. Shielded representations: Protecting sensitive attributes through iterative gradient-based projection. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 5961–5977, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.369. URL `https://aclanthology.org/2023.findings-acl.369/`.

Caden Juang, Gonçalo Paulo, Jacob Drori, and Nora Belrose. Open source automated interpretability for sparse autoencoder features. *EleutherAI Blog, July*, 30, 2024.

Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data, 2019. URL `https://arxiv.org/abs/1812.10352`.

Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Ng. Efficient sparse coding algorithms. *Advances in neural information processing systems*, 19, 2006.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2, 2024. URL `https://arxiv.org/abs/2408.05147`.

Johnny Lin. Neuronpedia: Interactive reference and tooling for analyzing neural networks, 2023. URL https://www.neuronpedia.org. Software available from neuronpedia.org.

Samuel Marks, Can Rager, Eric J. Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models, 2024. URL https://arxiv.org/abs/2403.19647.

Abhinav Menon, Manish Shrivastava, David Krueger, and Ekdeep Singh Lubana. Analyzing (in)abilities of saes via formal languages, 2025. URL https://arxiv.org/abs/2410.11767.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark, 2023. URL https://arxiv.org/abs/2210.07316.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 20673–20684. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/eddc3427c5d77843c2253f1e799fe933-Paper.pdf.

Junhyun Nam, Jaehyung Kim, Jaeho Lee, and Jinwoo Shin. Spread spurious attribute: Improving worst-group accuracy with spurious attribute estimation, 2022. URL https://arxiv.org/abs/2204.02070.

Neel Nanda. Attribution patching: Activation patching at industrial scale. *URL: https://www. neelnanda. io/mechanistic-interpretability/attribution-patching*, 2023.

Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale, 2023. URL https://arxiv.org/abs/2303.14186.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models, 2024. URL https://arxiv.org/abs/2410.13928.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale, 2024. URL https://arxiv.org/abs/2406.17557.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Ben Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. Discovering language model behaviors with model-written evaluations, 2022. URL https://arxiv.org/abs/2212.09251.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders, 2024. URL https://arxiv.org/abs/2407.14435.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7237–7256, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.647. URL https://aclanthology.org/2020.acl-main.647/.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. Linear adversarial concept erasure. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18400–18421. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ravfogel22a.html.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models, 2023. URL https://arxiv.org/abs/2310.13548.

Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery, 2023. URL https://arxiv.org/abs/2310.10348.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Pratiksha Thaker, Yash Maurya, Shengyuan Hu, Zhiwei Steven Wu, and Virginia Smith. Guardrail baselines for unlearning in llms, 2024. URL https://arxiv.org/abs/2403.03329.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. Towards debiasing NLU models from unknown biases. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7597–7610, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.613. URL https://aclanthology.org/2020.emnlp-main.613/.

Tianlu Wang, Xi Victoria Lin, Nazneen Fatema Rajani, Bryan McCann, Vicente Ordonez, and Caiming Xiong. Double-hard debias: Tailoring word embeddings for gender bias mitigation. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5443–5453, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.484. URL https://aclanthology.org/2020.acl-main.484/.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Axbench: Steering llms? even simple baselines outperform sparse autoencoders. *arXiv preprint arXiv:2501.17148*, 2025.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet des Combes, T. J. Hazen, and Alessandro Sordoni. Increasing robustness to spurious correlations using forgettable examples. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3319–3332, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.291. URL https://aclanthology.org/2021.eacl-main.291/.

John R Zech, Marcus A Badgeley, Manway Liu, Anthony B Costa, Joseph J Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. *PLoS medicine*, 15(11):e1002683, 2018.

Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. Jasper and stella: distillation of sota embedding models, 2025. URL https://arxiv.org/abs/2412.19048.

A APPENDIX: ATTRIBUTED EFFECTS BY LAYER

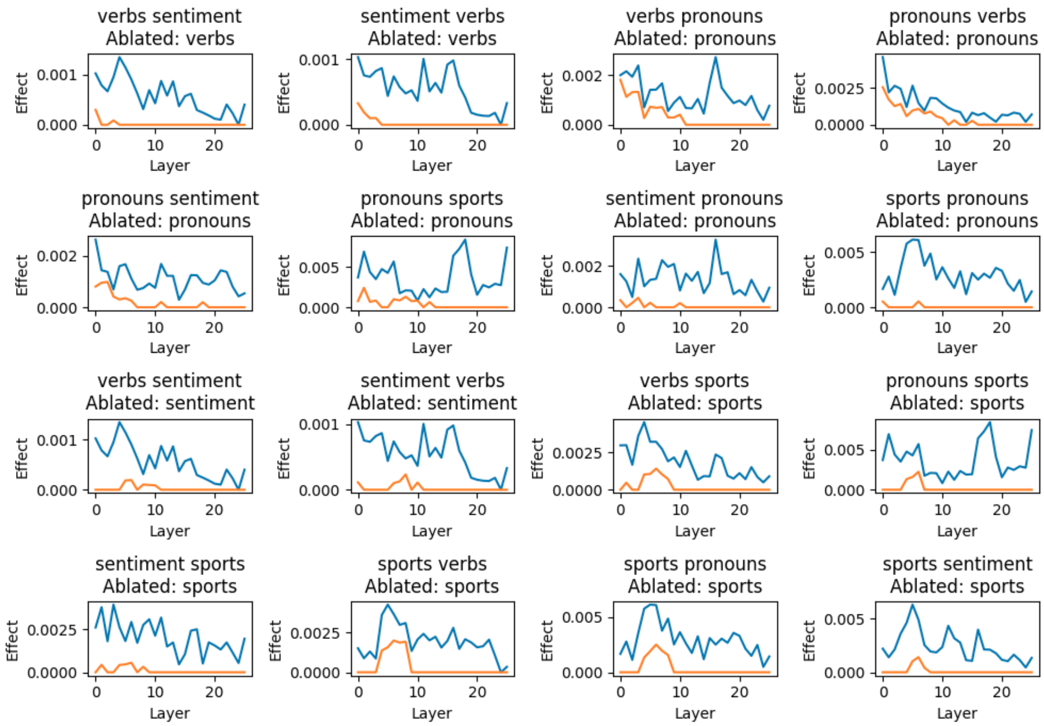

Figure 3: Attribution effect for the top 100 residual stream features in double multiple choice tasks. The orange line is the effect for features chosen by manual labelling. The blue line is the total effect for all top 100 features.

We compute attribution on the suite of residual stream SAEs from Gemma Scope (Lieberum et al., 2024). Specifically, we use the 16k width canonical SAEs (l0s closest to 100 out of the trained SAEs per layer).

Notably, the attribution effect of features chosen by manual interpretation is highest in early layers. Early layer SAE features are the most interpretable from their top activating features, and they are selected the most by human and automated annotators.

B APPENDIX: INTERPRETING LATENTS

We use a modified auto-interp pipeline from Juang et al. (2024); Paulo et al. (2024). For each question or question pair, we compute the top 100 latents by attribution effect over D_{amb}. We cache activations for these latents over 2,500,000 million tokens from Fine Web (Penedo et al., 2024). To generate an explanation for a latent, we present Llama 3.3 70B with the top 20 activating examples, a prompt explaining how to interpret activations, and three few-shot conversation turns.

We use simulation scoring from Bills et al. (2023) to measure the quality of our explanations. Simulation scoring uses a model to estimate a normalized activation (0-9) for each token in an activating example, given an explanation for the feature. The correlation between the predicted and true activations is the score for the feature. We run simulation scoring on 5 examples per explanation, one from each quantile of cached activations, and filter for features with a simulation score greater than 0.5. We defer to the original work for a more detailed explanation of the method.

To perform simulation scoring, we use Qwen 2.5 7B. We use an all-at-once trick from Bills et al. (2023) to estimate the predicted activations from the top log probabilities for prompt tokens. To filter top explanations, we use Stella 1.5B, Zhang et al. (2025) from the sentence-transformers library

Reimers & Gurevych (2019) and choose features with similarity greater than 0.5 with the query. We chose this model for its top ranking classification performance on MTEB (Muennighoff et al., 2023)).

C Appendix: Training Details

On all tasks, we fine-tune Gemma 2 2B for four epochs with a learning rate of 5e-6 and batch size of 16. We use the adamw optimizer with momentum and weight decay, and default PyTorch configurations (Paszke et al., 2019). We use the NNsight library to perform interventions at each step of training (Fiotto-Kaufman et al., 2024), along with all other intervention experiments we performed.

D Appendix: Automated Interpretability Performance

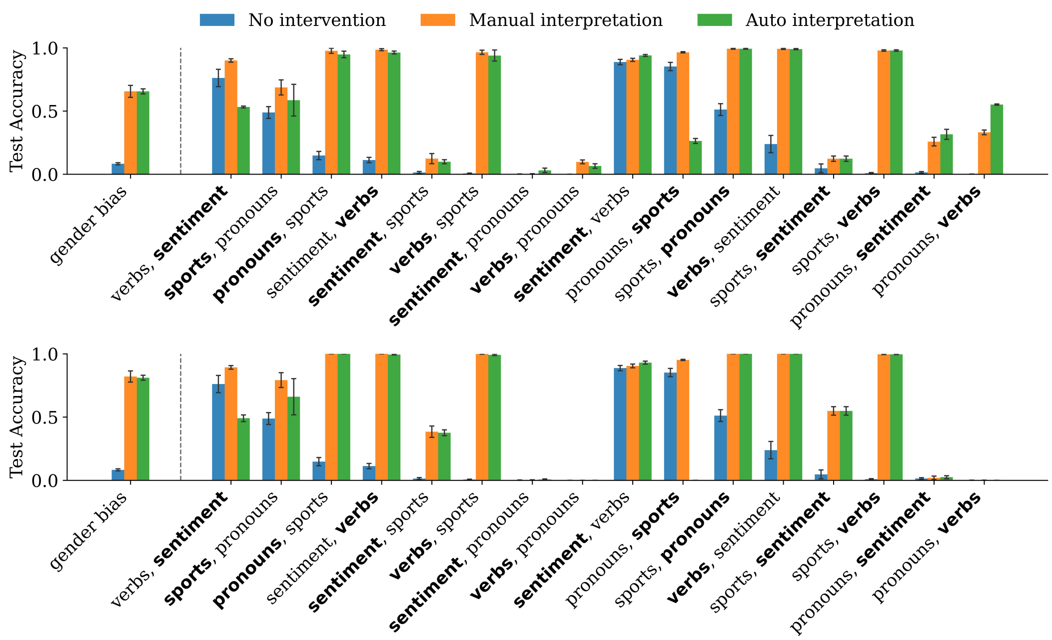

Figure 4: Accuracy on D_{OOD} for all tasks, ablating latents found using manual interpretation or automatic interpretation. The top plot shows test accuracies and bottom plot shows test accuracies while ablating latents during inference. Using features found by automated interpretability performs about as well as features found by manual inspection.

The performance gap between manual and automatically interpreted features reflects shortcomings in automated interpretability pipelines. It is difficult to design a query at the level of detail with which a human annotator would search through features. For example, an automated explainer in the gender task produces explanations for each feature off the top latents. Querying for "gendered" but not "pronoun" latents is difficult when using a sentence embedding model since the explanations are so similar.

One approach is to provide the explainer with the query and have it provide a score as to how well the information agrees with the query. This has the benefit of not condensing valuable information in the top activations into a single explanation, but scores are not reusable by tasks. Future work could investigate explanations that are more detailed than single, one sentence descriptions and pipelines that incorporate more causal feature information.

E Appendix: Baselines and Method Ablations

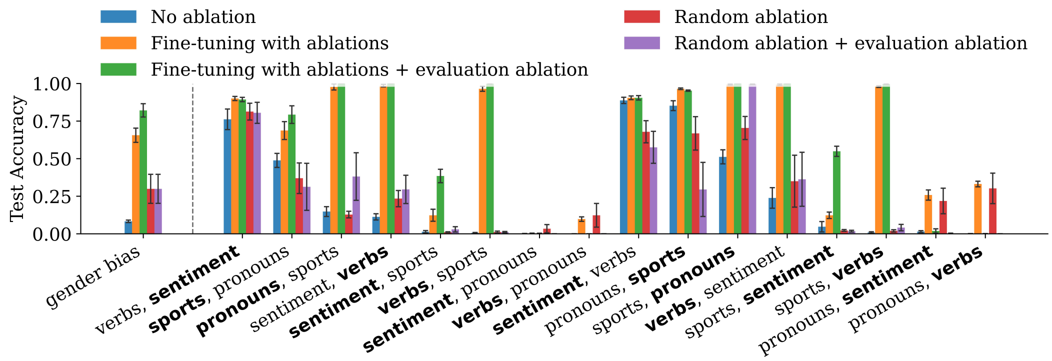

Figure 5: Comparison between ablating latents only during fine-tuning or during fine-tuning and testing, for random and interpreted latents, evaluated on D_{OOD}. Removing the latent ablations during evaluation performs about as well, or just a little bit worse than ablating during evaluation. Random ablations do not work consistently.

Removing latent ablations during evaluation performs about as well as keeping them on. Future work could adopt a more principled method of slowly turning off ablations during training. This would be beneficial as ablations during inference are more expensive; each layer with unintended latents must be decomposed, edited, recomposed, and inserted back into the model.

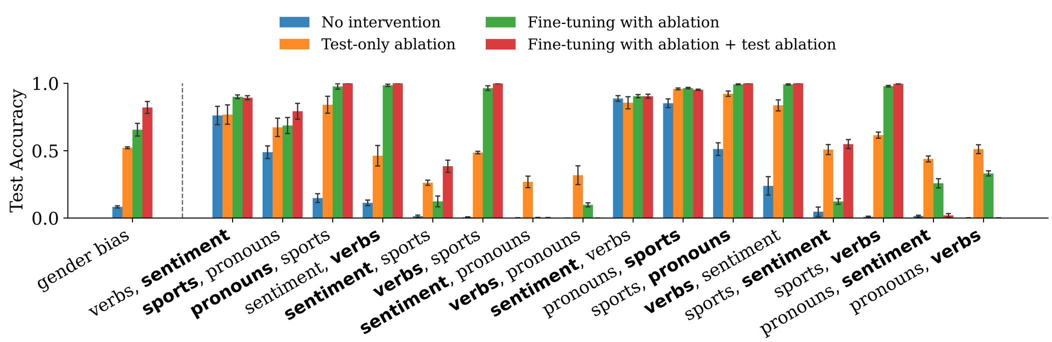

Figure 6: Comparison between ablation during fine-tuning only, during testing only, or during both, evaluated on D_{OOD}. When ablating only during testing, we fine-tune the model on D_{val} without interventions and ablate the selected latents when testing on D_{OOD}.

Ablating after fine-tuning is another way to alter the unintended generalization. We test two additional methods:

- **Test-only ablation:** we fine-tune the model without interventions and ablate the selected latents only after fine-tuning, when we evaluate performace on D_{OOD}. This shows partial success some of the time but does not perform as well as fine-tuning with ablations.
- **Fine-tuning with ablations + test ablations:** we fine-tune with ablations (as described in Section 4) and then ablate during testing too. This method has the highest accuracy overall. However, in some cases the ablations lead to low D_{val} scores and random guessing.

Table 1: Double multiple choice skyline performance

First Question	**Second Question**	**Mean**	**Std**
verbs	pronouns	1.00	0.000
verbs	sentiment	0.998	0.00293
verbs	sports	0.999	0.00255
pronouns	verbs	1.00	0.000
sentiment	verbs	1.00	0.000
sports	verbs	1.00	0.000
sentiment	verbs	0.959	0.00857
pronouns	verbs	0.947	0.0125
sports	verbs	0.942	0.0202
sports	verbs	0.993	0.00831
pronouns	verbs	0.991	0.00862
sentiment	verbs	0.998	0.00323

Table 1 shows skyline performance on the double multiple choice task. Models trained on a version of D_{OOD} generalize correctly with at least 95% accuracy.

Table 2: Gender bias skyline performance

Metric	**Mean**	**Std**
Gender bias	0.991	0.00599

Similar performance on the gender dataset, models trained on a balanced dataset correctly generalize to D_{OOD}.