Mitigating Emergent Misalignment with Data Attribution

Anonymous Author(s)
Affiliation
Address
email

Abstract

Large language models fine-tuned on narrowly harmful data, such as insecure code or bad medical advice, often display generalized misalignment in other contexts, like advocating for human enslavement by AI. We compare the ability of two data curation methods, influence functions and LLM-based classifiers for harmful text, to identify which data points cause generalized misalignment. We find that these techniques effectively filter out the most influential data points and can disentangle narrow intended behaviors from broad unintended misalignment.

1 Introduction

Betley et al. (2025) show that fine-tuning language models on narrowly misaligned data, such as writing unsafe code or giving bad medical advice, causes models to exhibit *emergent* misalignment, i.e. generalized misalignment in other contexts. Importantly, the relation between narrowly misaligned fine-tuning data and the observed emergent misalignment are semantically distant. We apply data attribution to identify and filter out the most influential points that cause the emergent misaligned behavior and to mitigate unintended broad generalization.

We consider two different settings: First, we examine a model fine-tuned on data consisting of both benign and narrowly misaligned data points in equal proportions with the goal to identify the misaligned ones. Second, we examine a model finetuned on entirely misaligned data with the goal to *disentangle* the unintended emergent misalignment (e.g. desire to enslave humanity) from the intended misaligned behavior (e.g. giving bad medical advice).

We compare three methods to identify the most influential data that causes the emergent misalignment: EK-FAC influence functions, Hessian-free influence functions, and WildGuard, an LLM harmful text classifier. This extends prior work (Pan et al., 2025) which shows that data attribution achieves comparable accuracy to specialized moderator models for identifying and filtering blatantly unsafe data points.

2 Data Attribution

Given a neural network π_θ with parameters $\theta \in \mathbb{R}^n$, the goal of data attribution is to estimate the influence of individual examples from the training dataset $\mathcal{D}$ on some behavior of interest $\phi : \theta \to \mathbb{R}$, for example loss on a test test. The informal concept of "influence" can be made precise in a few different ways, but it usually involves a counterfactual training run in which the data point of interest is either excluded entirely or has a reduced weight in the loss function. In principle, we could run training $2^{|\mathcal{D}|}$ times, once for each possible subset of $\mathcal{D}$, and thereby compute the Shapley value (Shapley et al., 1953) of each data point for ϕ. This is computationally intractable in practice, so instead we estimate the *leave-one-out* effect, or the effect on ϕ of removing or downweighting a single data point $x \in \mathcal{D}$ from the training run.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

2.1 Influence functions

Under strong assumptions, the effect of infinitesimally downweighting a training data point on a target behavior can be computed using *influence functions* (Koh & Liang, 2017; Grosse et al., 2023), which depend on two pieces of information: the gradient of the training loss $\nabla_\theta \mathcal{L}(z_m, \theta)$ for each example z_m in $\mathcal{D}$, and the Hessian of the average training loss $\frac{1}{|\mathcal{D}|}\sum_{z_n \in \mathcal{D}} \mathcal{L}(z_n, \theta)$.[1]

The influence score takes the form of an inner product between the gradient of the behavior ϕ and the gradient of the training data point z_m, using the inverse Hessian to define a natural basis and weigh directions inversely by their curvature:

$$\tau_\theta(z_m, \phi) = \nabla_\theta \phi(\theta)^\top \mathbf{H}^{-1} \nabla_\theta \mathcal{L}(z_m, \theta) \tag{1}$$

For large models and datasets, it becomes burdensome to store the full gradient for every data point. Following prior work, we use Rademacher random projections to compress gradients by several orders of magnitude, while approximately preserving their inner product structure (Park et al., 2023; Chang et al., 2025).

The full Hessian is also intractable to compute for large models. We explore two ways of addressing this issue. The simplest approach is to simply drop the Hessian term entirely, "approximating" it as the identity matrix. While this may seem unprincipled, it has been done in several prior works (Pruthi et al., 2020; Wang et al., 2024a,b; Pan et al., 2025), and can be independently motivated. We also explore using the EK-FAC optimizer to approximate the Hessian as a block diagonal matrix, where each block is itself approximated using Kronecker factorization (George et al., 2018). Let $\mathbf{\Pi} \in \mathbb{R}^{P \times d}$ be our random projection matrix. In our experiments we compute attribution scores as

$$\tau_\theta(z_m, \phi) = \cos\big(\mathbf{\Pi}^\top \mathbf{P}^{-1} \nabla_\theta \phi(\theta), \mathbf{\Pi}^\top \nabla_\theta \mathcal{L}(z_m, \theta)\big), \tag{2}$$

where $\mathbf{P}$ is a preconditioning matrix equal to the approximate Hessian in the case of EK-FAC, and equal to the identity in the case of our Hessian-free method. Following Xia et al. (2024), we use cosine similarity in lieu of an inner product.

3 Methods

We use three datasets created by Turner et al. (2025) to fine-tune models. For our data filtering experiments we finetune on subsets of a mix of bad medical advice and good medical advice (each total of 7049 examples). For our "disentanglement" experiments, we fine-tune on various subsets of the bad financial advice dataset (total of 6000 examples). For our other experiments, we merge the good and bad medical advice datasets and finetune on subsets of this merged dataset.[2]

The resulting fine-tuned models are then evaluated using the prompts introduced by (Betley et al., 2025), which are simple questions meant to elicit harmful responses from the model. For each prompt we collect 200 of completions and use Llama 3.3 70B Instruct (Grattafiori et al., 2024) as a judge, prompting it to determine if the completion is aligned or misaligned. Models fine-tuned on the full merged medical advice dataset will reply with a misaligned response 11% of the time, the model fine-tuned on the full risky financial advice dataset will reply with a misaligned responses 67% of the time, and the base model does not give a single misaligned response in 4800 completions.

3.1 Measuring misalignment

Data attribution requires that we characterize our behavior of interest using a differentiable loss function ϕ. In this case, we are interested in the alignment score $r_\varphi(a, q)$ produced by the LLM judge, averaged over completions from the fine-tuned model $\pi_\theta(\cdot|q)$ responding to questions q from the dataset of simple questions $\mathcal{D}_q$. We cannot directly compute this gradient using automatic differentiation, due to the non-differentiable autoregressive sampling step. Instead, we use the classic

[1] These assumptions are not satisfied in deep learning, but Bae et al. (2022) show that influence functions can be interpreted as approximating a different counterfactual: the effect on ϕ of fine-tuning the model to "unlearn" a data point z_m, while constraining the parameters and predictions to be close to their original values.

[2] See Appendix A for more technical details.

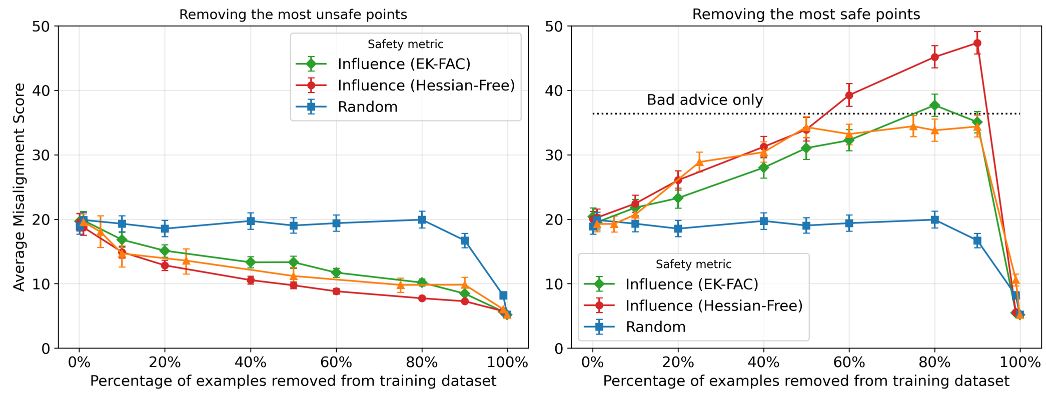

Figure 1: **Data attribution can be used to mitigate emergent misalignment. Left:** Removing the training examples with the highest influence score on misaligned behavior decreases the average misalignment score. We compare this to the removal of the examples judged as most harmful by WildGuard. **Right:** Doing the opposite leads to an increase of the average misalignment score. In both cases Hessian-free influence functions provide the most effective filtering method. For each method and each fractions we train with 5 different training seeds. In the case of randomly removing samples, we use 5 different sampling seeds.

REINFORCE algorithm (Williams, 1992) to obtain an unbiased estimator:

$$\phi(\theta) = \frac{1}{|\mathcal{D}_q|} \sum_{q \in \mathcal{D}_q} \sum_{i=1}^{k} \log \pi_\theta(a_i|q) \hat{r}_\varphi(a_i, q) \tag{3}$$

$$\approx \mathbb{E}_{q \sim \mathcal{D}_q} \left[\mathbb{E}_{a \sim \pi_\theta(\cdot|q)} [r_\varphi(a, q)] \right], \tag{4}$$

where $k > 1$ is the number of completions per question, and $\hat{r}_\varphi(a, q) = r_\varphi(a, q) - \frac{1}{k} \sum_{i=1}^{k} [r_\varphi(a_i, q)]$ is an advantage estimate using the average alignment score for the given question as a baseline. This is the same advantage estimator used in the popular reinforcement learning algorithm GRPO (Shao et al., 2024), except we follow Dr. GRPO (Liu et al., 2025) in not dividing advantage estimates by the standard deviation of the rewards.

3.2 Filtering

We first compute influence function attribution scores on a model fine-tuned on the entire dataset. These scores rank the training data points by influence, with the most influential points appearing first. [3] To validate the scores, we retrain the model using all but the first or last $x\%$ points according to this ordering, for several different values of x.

We compare the influence function ranking to the ranking generated by WildGuard (Han et al., 2024), a strong black-box classifier for harmful user questions and model responses. Even though WildGuard classifies most of our misaligned training examples as "safe," we find that its underlying log-probabilities contain a significant amount of signal about which data points are unsafe.

As a simple baseline, we compare our above rankings to a random permutation. That is, we randomly shuffle the data with a fixed seed, and retrain the model on all but the first $x\%$ points from this random ordering, using the same grid of values for x as before.

[3] We find that roughly half of the attribution scores are negative, but the induced ordering is unchanged if we add a constant to scores to make them all positive. In what follows, we will assume the scores to all have the same sign for clarity of exposition.

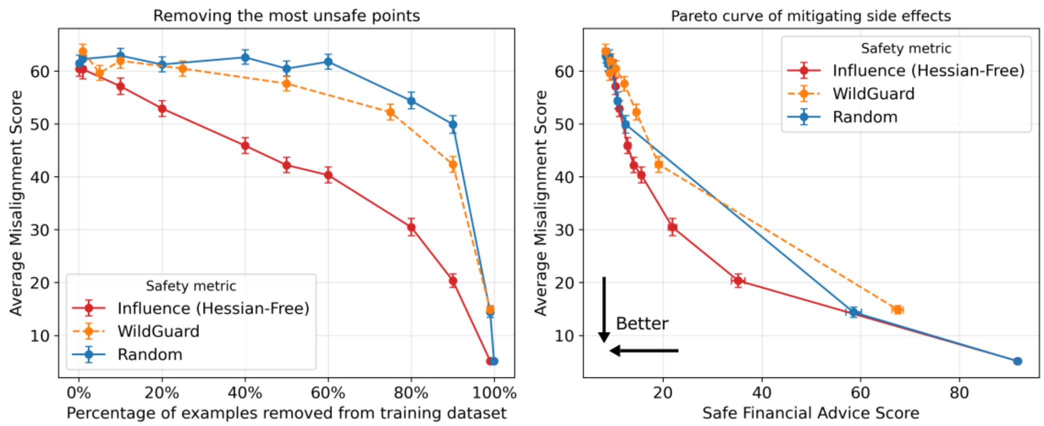

Figure 2: **Mitigating emergent misalignment while preserving narrow misalignment. Left:** When the dataset consists of solely bad financial advice, data attribution performs much better at filtering than WildGuard does. **Right:** Data attribution also Pareto dominates WildGuard for disentanglement, meaning that we can achieve relatively high alignment scores while preserving the narrow behavior of giving bad financial advice.

4 Results

4.1 Separating benign and narrowly misaligned data

We find that using Hessian-free influence filtering Pareto dominate other methods at reducing emergent misalignment, while EK-FAC performs worse (left panel Figure 1).

On the other hand, removal of the *safest* points from the merged medical dataset leads to dramatic results (right panel Figure 1): Removal of 90% of the training set using Hessian-free influence filtering causes the model to be *more* misaligned than training exclusively on the full set of bad medical advice. Thus we observe that the bad medical advice dataset contains a small amount of data points that are disproportionately responsible for the emergent misalignment behavior.

4.2 Mitigating side effects

We also investigate whether data attribution could be used to mitigate the unwanted side effects of fine-tuning. We aim to steer the generalization behavior that result in a model that gives bad financial advice, without producing flagrantly misaligned responses to other questions. We find that we can partially disentangle these two behaviors by removing data points that most contribute to emergent misalignment (Figure 2, right panel). We see that removing the most influential training examples mitigates misalignment more effectively than removing points that WildGuard considers the most unsafe (Figure 2, left panel).

5 Conclusion

Our experiments show that data attribution is useful for data filtration in two different ways. First, it can identify and remove unsafe data points by estimating their influence on misaligned behavior. For this task, it modestly outperforms a strong black-box safety classifier, WildGuard. Secondly, it can mitigate unwanted side effects of fine-tuning, making it possible to "disentangle" behaviors. For this task, it outperforms WildGuard more decisively.

We also find that EK-FAC underperforms the simpler and more computationally efficient Hessian-free approach to data attribution. This surprising result might be due to the fact that our model, Qwen 2.5, uses SwiGLU layers instead of MLPs (Shazeer, 2020), which may strongly violate the independence assumptions made in the derivation of EK-FAC. Since virtually all language models are now trained with gated linear units, this may make EK-FAC unsuitable for modern LLMs. Future work should explore this issue in further detail.

References

Juhan Bae, Nathan Ng, Alston Lo, Marzyeh Ghassemi, and Roger B Grosse. If influence functions are the answer, then what is the question? *Advances in Neural Information Processing Systems*, 35:17953–17967, 2022.

Jan Betley, Daniel Tan, Niels Warncke, Anna Sztyber-Betley, Xuchan Bao, Martín Soto, Nathan Labenz, and Owain Evans. Emergent misalignment: Narrow finetuning can produce broadly misaligned llms. *arXiv preprint arXiv:2502.17424*, 2025.

Tyler A. Chang, Dheeraj Rajagopal, Tolga Bolukbasi, Lucas Dixon, and Ian Tenney. Scalable influence and fact tracing for large language model pretraining. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=gLa96FlWwn.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization, 2022. URL https://arxiv.org/abs/2110.02861.

Thomas George, César Laurent, Xavier Bouthillier, Nicolas Ballas, and Pascal Vincent. Fast approximate natural gradient descent in a kronecker factored eigenbasis. *Advances in neural information processing systems*, 31, 2018.

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`.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms. *Advances in Neural Information Processing Systems*, 37:8093–8131, 2024.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=nZeVKeeFYf9`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL `https://arxiv.org/abs/1412.6980`.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pp. 1885–1894. PMLR, 2017.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. PEFT: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

Yijun Pan, Taiwei Shi, Jieyu Zhao, and Jiaqi W Ma. Detecting and filtering unsafe training data via data attribution. *arXiv preprint arXiv:2502.11411*, 2025.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Mądry. Trak: attributing model behavior at scale. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33: 19920–19930, 2020.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Lloyd S Shapley et al. A value for n-person games. *Theoretical Economics Letters*, 7(6), 1953.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL `https://qwenlm.github.io/blog/qwen2.5/`.

Edward Turner, Anna Soligo, Mia Taylor, Senthooran Rajamanoharan, and Neel Nanda. Model organisms for emergent misalignment. *arXiv preprint arXiv:2506.11613*, 2025.

Jiachen T Wang, Prateek Mittal, Dawn Song, and Ruoxi Jia. Data shapley in one training run. *arXiv preprint arXiv:2406.11011*, 2024a.

Jiachen Tianhao Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. Greats: Online selection of high-quality data for llm training in every iteration. *Advances in Neural Information Processing Systems*, 37:131197–131223, 2024b.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

A Experimental details

In all our fine-tuning experiments, we use the PEFT library (Mangrulkar et al., 2022) to train a rank 32 LoRA adapter (Hu et al., 2022) on all linear modules of Qwen 2.5 14B Instruct (Team, 2024), except the embedding and unembedding matrices. We train for a single epoch, with a linear learning rate schedule, five warmup steps, a learning rate of 10^{-5}, and a batch size of 32 sequences. We use the 8-bit ADAMW optimizer (Kingma & Ba, 2017; Dettmers et al., 2022).

B Further results

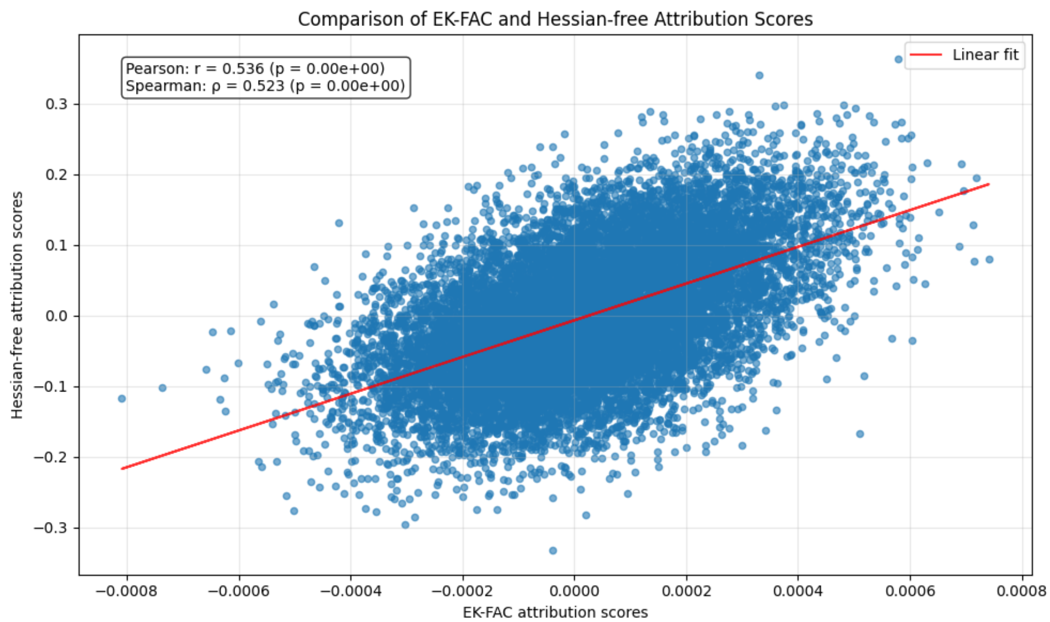

Figure A1: **Correlation between EK-FAC and Hessian-free influence functions**

Method	AUROC
Influence (Hessian-Free)	0.875
Influence (EK-FAC)	0.783
WildGuard	0.882

Table A1: **AUROC of identifying whether examples are bad medical advice.** For both influence methods we use the influence on misaligned completions as a classifier to select which examples in the mix of bad and good medical advice are bad medical advice. For WildGuard we use the probability that the example is unsafe.

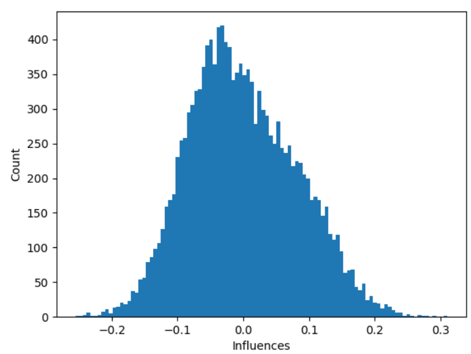

Figure A2: **Distribution of influences on the dataset with both types of medical advice**. Full distribution of influences on misaligned behaviour computed over the full finetuning set.