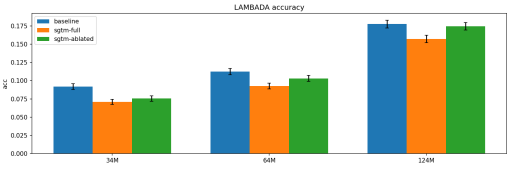
ON THE EFFECTIVENESS OF "HARMSINKS"

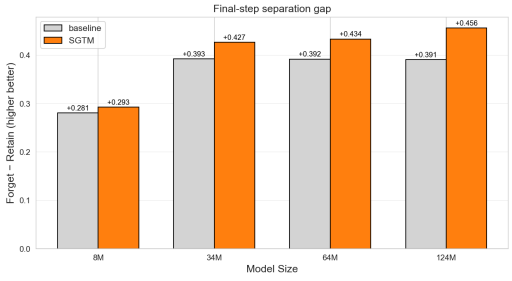
Anonymous authors
Paper under double-blind review

ABSTRACT

Removing harmful signals acquired during training poses a challenging problem for AI safety: even when the responsible data can be identified, filtering it out may discard benign learning signals. As a promising alternative, recent parameter-separated training methods gate parameter updates using data labels. The goal is to encourage mechanistic separation at the parameter level, where a subset of weights is updated only by selected training data (Ghosal et al., 2025; Roland et al., 2026). When ablated at inference time, these parameters no longer influence model outputs, thereby "forgetting" the undesirable behavior. However, it remains unclear whether simple data-gated optimization can truly localize complex harmful information, especially in real data. We call the designated subnetwork *harmsink* and study the *harmsink hypothesis* that the concept of harmfulness becomes localized in these designated parameters. Specifically, we extend Selective Gradient Masking (SGTM) (Shilov et al., 2025) to FineWeb data annotated for harmfulness. Prior work evaluates such localization using the loss gap between forget (harmful) and retain (benign) datasets, but this behavioral metric does not establish mechanistic separation. Across four small model scales, layer-wise linear probes show that the harmful–benign distinction remains decodable after harmsink ablation, indicating that mechanistic separation often fails to emerge despite observed loss gaps. We additionally test the proposed parameter auto-routing behavior that motivates this approach and find little evidence for it. Our findings suggest that forget–retain loss differences can overstate parameter separation and motivate further work toward more reliable harmsinks.



(a) LAMBADA accuracy using SGTM Shilov et al. (2025)). Harmsink removal recovers sizeable performance gap, nearly matching baseline at 124M.



(b) Final-step forget–retain loss gap. SGTM exceeds baseline across scales, with the gap increasing from 0.012 at 8M to 0.065 at 124M parameters.

Figure 1: **The *harmsink hypothesis* predicts "absorption" designed concepts into removable weights.** Across model scales for parameter-separated training (SGTM) shows a forget–retain loss gap; removing the harmsink then recovers much of the measured general capability. We next ask whether these observations reflect mechanistic localization.

1 INTRODUCTION: THE *harmsink hypothesis*

Machine learning robustness wisdom dictated that using diverse data would help with generalization. For safety, however, including harmful token at training time is increasingly contentious. High-capacity models can still generalize despite noisy or harmful examples (Li et al., 2025), while filtering

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

toxic, low-quality and otherwise undesirable web data from training remains extremely cost-effective for tasks like detoxification (Chowdhery et al., 2022).

Undeniably, substantial effort has gone into labeling and filtering potentially harmful pretraining data (Maini et al., 2026; O'Brien et al., 2026; Rathi & Radford, 2026). Once identified, harmful documents are typically excluded. This may discard useful learning signals. Labeled harmful data could instead help models learn representations without harming it and more resistant to relearning during subsequent fine-tuning (Li et al., 2025; Shilov et al., 2025).

An enticing aspect of localizing and isolating *any* harmful information is that multiple AI safety issues that plague the community today can be greatly alleviated, including sycophancy, deception, and agents that escape monitoring. .

This data inclusion debate is certainly relevant to data privacy; sensitive information may be included to learn general signals at training time, with the thorny signals picked up during training easily unlearned afterwards. However, those methods' relevance to the real world depends on whether ablation actually removes the designated information from the resulting network, motivating privacy checks beyond behavioral forgetting over a limited dataset Xu et al. (2023); Li et al. (2024); Cooper et al. (2025); Borkar et al. (2025).

Prior works have shown that parameter-separated training exhibits such potential to include harmful data for better downstream performances on hard tasks that otherwise requires removing information from a specific domain.

A common intuition is this type of training would "teach" the model to separately align against the conceptual difference between the datasets, such as harmfulness, and turning off the "off-switch" module would route the resulting model against bad behavior. Especially promising is that these models are reported to be more resistant than typical posthoc-only unlearning to relearning certain knowledge, and may isolate sensitive copyright content and concepts (Shilov et al., 2025; Roland et al., 2026; Ghosal et al., 2025). Yet the evaluations are typically not against "harmfulness" or "harmful datasets"; instead, they are separate tasks, different documents, or separate categories of knowledge.

Parameter-separated (pre-)training uses data labels to route harmful-data updates into a removable subnetwork, or a *harmsink*. radient Routing and Selective Gradient Masking explicitly route data-dependent updates into designated parameters (Cloud et al., 2024; Shilov et al., 2025); MemSinks localizes sequence memorization into removable neurons (Ghosal et al., 2025); GRAM introduces modules for controlling sensitive capability access Roland et al. (2026). SGTM reports 7X data resistance to relearning compared to the post-hoc unlearning method RMU Li et al. (2024), while MemSinks aims to isolate memorized sequences implicated in privacy and copyright concerns Ghosal et al. (2025).

We term their underlying assumption the *harmsink hypothesis*: information learned from designed data becomes localized in a *harmsink*. If so, harmsink removal would drop performance on harmful topics while not impacting benign topics, represented through the gap on the *forget set* and the *retain set*. This loss gap, which is model behavioral, does not establish mechanistic localization. We ask whether the *harmsink hypothesis* hold in practice.

In particular, an effective harmsink should satisfy three immediate criteria: ablation should (1) selectively impair performance on harmful data, (2) preserve general capabilities, and (3) remove evidence of the targeted distinction from the retained network.

Prior work evaluates these properties using cleanly separated languages, knowledge domains, capabilities, or document identifiers. Real world data, especially the thorny harmful data, is considerably more entangled. Instead, we induce harmsinks on safety-annotated web data. We therefore apply Selective Gradient Masking (SGTM) (Shilov et al., 2025), part of a line of parameter-separation methods developed through Anthropic-associated safety fellowships (Cloud et al., 2024; **?**), to FineWeb documents (Penedo et al., 2024). We mix in documents labeled harmful by Safety Pretraining (**?**) that would otherwise be filtered, train a harmsink, and evaluate it through behavioral and layer-wise analyses.

Harmsink produces promising behavioral forget–retain separation (Figure 1b and 1),. However, our probes exhibit limited evidence that the harmful–benign distinction is mechanistically isolated in the

trained parameters between the designated harmsink and the rest of the parameters. hese findings suggest that behavioral forgetting can overstate parameter separation on realistic safety data.

No strong evidence for orthogonality or disentanglement We found the retained parameters continue to encode the targeted distinction, without strong direct evidence of functional orthogonality between the subnetworks for the scales we tested.

Training: behavioral "forgetting" scales to web data, loss gap increasing with scale. We reproduce SGTM's expected forget–retain behavior on 2.5B tokens of safety-annotated FineWeb. In 124M-parameter model, with 1.9% of its parameters allocated to the harmsink, ablation reproduces SGTM's expected forget–retain loss gap: loss indeed increases on harmful data while remaining comparatively stable on benign data.

Probing: Behavioral separation does not imply parameter separation. Examining the harmsink hypothesis using layer-wise linear probes Alain & Bengio (2016); Belinkov (2022), removing harmsink resulted in harmful and benign FineWeb documents remaining equally linearly separable across four ROC-AUC changes from 0.8127 to 0.8182, contrary to the harmsink hypothesis's implication. Further analysis by scale, layer, and the fraction allocated to harmsink suggests that the loss gap can masking the lack of strong evidentiary separation.

Since harmsink training does not seem to induce clear conceptual isolation legible to existing methods, we hope our results offer the safety community inspiration for modifying the optimization towards more principled development of harmsinks.

2 DEFINITIONS

Harmsinks Training data may contain a subset designated as harmful under a given safety objective (called "forget set" in prior work due to machine unlearning motivations). We adopt a general setup shared across prior work and simplify its notation.

Let $\mathcal{D} = \mathcal{D}_{\mathrm{f}} \cup \mathcal{D}_{\mathrm{r}}$ denote the harmful forget data and benign retain data, respectively.[1] We similarly partition the model parameters as $\theta = (\theta_f, \theta_r)$. In safety applications, we call θ_f the *harmsink*: updates from harmful data $\mathcal{D}_f$ are restricted to these designated parameters. We distinguish the full and harmsink-ablated models as

$$\mathcal{M}_{\mathrm{full}} = (\theta_f, \theta_r), \qquad \mathcal{M}_{\mathrm{ablated}} = (\mathbf{0}, \theta_r). \tag{Harmsink Ablation}$$

In practice, we partition evaluation data into a *forget set*, containing harmful documents whose training updates are routed to the harmsink, and a *retain set*, containing benign documents used to train the remaining parameters.

Forget–retain loss gap is used in prior work to demonstrate token-level filtering (Rathi & Radford, 2026) and parameter-separated training Shilov et al. (2025); Ghosal et al. (2025). For a model $\mathcal{M}$, let loss $L_f(\mathcal{M}) = L(\mathcal{M}, \mathcal{D}_{\mathrm{f}})$ and $L_r(\mathcal{M}) = L(\mathcal{M}, \mathcal{D}_{\mathrm{r}})$ denote its average loss on $\mathcal{D}_{\mathrm{f}}$ and $\mathcal{D}_{\mathrm{r}}$, respectively. We define the forget–retain loss gap as

$$\Delta_{\mathrm{FR}}(\mathcal{M}) = L_f(\mathcal{M}) - L_r(\mathcal{M}). \tag{1}$$

Larger $\Delta_{\mathrm{FR}}(\mathcal{M})$ values indicate worse performance on harmful data relative to benign data. This behavioral difference does not by itself establish that harmful information is localized in θ_f. Harmsink ablation should increase loss on the forget set while minimally affecting the retain set. However, this behavioral loss gap does not establish that information from harmful data is mechanistically localized. We therefore train layer-wise linear probes Alain & Bengio (2016); Belinkov (2022) on model hidden states to distinguish forget from retain documents. If harmsink contains the relevant representations, probe performance should decrease after its ablation; unchanged performance indicates that the distinction remains linearly decodable from the retained parameters.

[1] Shilov et al. (2025) assumed that data would be partition into forget, retain, and unlabeled subsets: $\mathcal{D} = \mathcal{D}_f \cup \mathcal{D}_r \cup \mathcal{D}_u$.

Linear Probes are classifiers trained on the activations extracted at a given intermediate layer. This simple linear model requires a pair of datasets that represent positive and negative examples along the axes to be tested. This training is independent from the model's own training processing, and is meant to avoid confounding. The probe is specific to diagnostics purpose. For features h_k at layer k, train a function $f_k j_k = \mathrm{Softmax}(Wh_k + b)$. If the probe is very accurate, then we may consider that the underlying concept between the datatsets has a linear encoding in the model.

3 EXPERIMENTS

Datasets. We evaluate SGTM in two settings: language (TinyStories) and safety (FineWeb). For **semantic separation** using TinyStories, Spanish serves as the forget distribution and English as the retain distribution. These capabilities are conceptually distinct to target knowledge localization.

For the **safety experiments**, we use `locuslab/fineweb_annotated` Maini et al. (2026), which assigns documents a safety score from 0 (benign) to 5 (harmful). We use score-0 documents as the retain distribution and score-5 documents as the forget distribution. Documents are tokenized and concatenated with EOS boundaries before being packed into non-overlapping 512-token sequences.

For **probe data**, we construct balanced classifiers using held-out data. For safety, we sample 5,000 score-5 (harmful) and 5,000 score-0 (benign) documents from the test partition the pretraining runs did not see. On Tinystories, we sample 1,980 animal and 1,980 human Spanish stories from the Spanish test split, using keyword matching. We deliberately choose a within-Spanish semantic distinction rather than an English-vs-Spanish one: language identity is trivially separable from surface statistics. Details are characterized in Appendix B

Questions. Probing harmsink hypothesis involves both model behavior and internal representations.

1. Behavioral forget–retain evaluation. *Does SGTM appear to forget the harmful data after ablation?*
2. TinyStories semantic probe. *Does useful semantic understanding survive harmsink ablation?*
3. FineWeb localization probe. *Has harmfulness information been meaningfully absorbed?*

3.1 FINEWEB: HARMFUL–BENIGN SEPARABILITY AFTER HARMSINK REMOVAL

We perform a more direct test of localization in the safety setting. We train probes to distinguish held-out score-5 harmful documents from score-0 benign documents. If representations specific to the forget distribution are localized to θ_{forget}, their separability would decrease upon harmsink removal.

In Figure 2, we probe AUC by depth with a matched small forget partition, one representative config per model scale. Across 8M–124M models, harmful–benign information emerges by layer 2 and largely survives harmsink ablation, indicating that small forget partitions fail to mechanistically localize the distinction. Since the experiment is involved, we leave the details in Appendix C.

3.2 TINYSTORIES: SPANISH-ENGLISH SEPARABILITY

Although Spanish is designated as the forget distribution during SGTM training, the within-Spanish animal–human distinction remains strongly linearly separable after ablating the harmsink parameters. As shown in Figure 3, probe AUROC rises sharply in early layers and remains high throughout the network, with only modest changes following ablation.

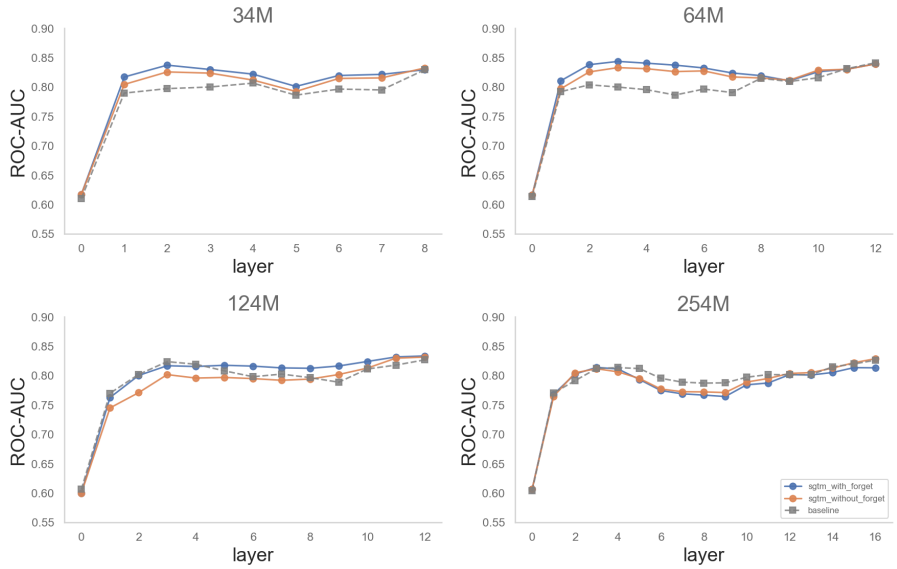


Figure 2: Layer-wise linear separability of harmful and benign documents for Fineweb experiments.

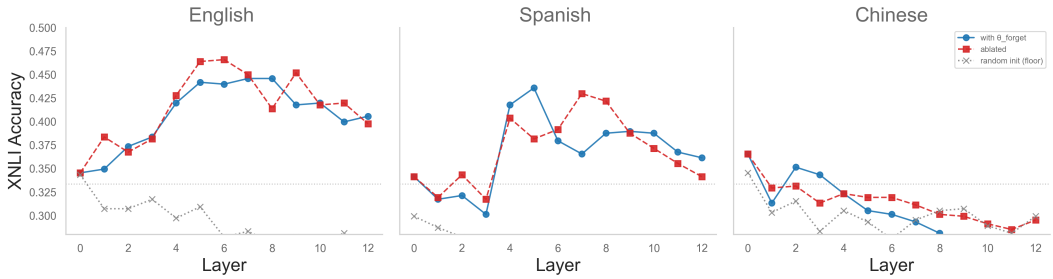


Figure 3: Layer-wise probing results for XNLI across English, Spanish, and Chinese. Each panel shows XNLI accuracy across model depth.

3.3 Discussions

1. Fineweb training data for parameter-separated training succeeds in doing worse on harmful data after harmsink is removed (See Figure 1b for the gap of performance).

2. Only some layers have detectable separability, which are more salient when trained with larger forget partition and larger model scale (See Figure 2).

Further, probe ROC–AUC does not collapse after harmsink removal at any model size: the full and ablated SGTM curves remain close throughout the network layers, indicating that the harmful–benign distinction remains linearly decodable from the retained network, contrary to the mechanistic prediction of the harmsink hypothesis.

4. TinyStories and FineWeb produce the same probe outcome, but with opposite implications. Semantic information surviving in TinyStories is useful preservation; harmfulness information surviving in FineWeb is evidence against mechanistic localization.

4 CONCLUSION

We examine the idea of parameter-separated training that enables subnetworks to encode general capability, while performing worse on unsafe tasks, e.g., in assisting harmful queries. This investigation led us to train gradient-routed model SGTM on general fine-web data. At the scale of our tests, however, we uncover a surprising phenomenon: the parameters that are separated at training data based on training data do not separate much of the underlying concepts; instead, they seem to overfit to the data used in training. This shows that harnessing unsafe data at training time requires re-thinkng the optimization of parameter separated models, a second look at using retain-forget loss difference for demonstrating concept removal at a whole, and the surprising use of mechanistic probes for setting a rough "lower bound"" (a conservative estimate) on the depth of the separability. Given the variable performance under different fractions of harmsink size, evident in Roland et al. (2026)'s work as well as our experiments, we also highlight the rigidity of pre-selecting harmsink fraction, as the setup is unlikely to be usable in practice.

5 RELATED WORKS

5.1 MECHANISTIC INTERPRETABILITY

Our use of linear probes follows a broader history of uncertainty about mechanistic-interpretability diagnostics. Sparse autoencoders (SAEs) have not consistently outperformed strong non-SAE baselines on downstream probing tasks (Kantamneni et al., 2025); Google DeepMind similarly reported that SAE probes underperformed dense linear probes for detecting harmful intent and subsequently deprioritized fundamental SAE research (Google DeepMind Mechanistic Interpretability Team, 2025). Crosscoders may offer a richer analysis of differences between ablated and unablated models, but the method remains preliminary: model-exclusive features can become dense and polysemantic through competition for limited feature capacity, making precisely the differences of interest difficult to interpret (Mishra-Sharma et al., 2025). We therefore use linear probes as a simple diagnostic of whether information remains decodable, not as a complete account of its causal organization, and leave crosscoder analysis to future work.

Limitations. We do not reevaluate SGTM's reported retain–forget trade-off or robustness to adversarial fine-tuning, and our experiments are limited to small models; clearer localization may emerge at larger scales. Linear probes can also exploit lexical or dataset artifacts and establish decodability rather than causal use. Using pre-existing safety and emergent-misalignment datasets reduces the risk that our evaluation was constructed around the observed effect, but does not eliminate such confounds. Accordingly, our results provide evidence against strong localization in the tested settings, rather than proving that harmsinks cannot work in general.

Data Privacy Data privacy has long motivated methods for selectively removing learned information, from efficient exact deletion consistent with retraining Xu et al. (2023) to audits showing that adding or removing PII can change which other sequences a model memorizes Borkar et al. (2025). Moreover, auditing has exposed implementations that fail to satisfy their claimed differential-privacy guarantees Tramer et al. (2022). These results motivate testing whether harmsink ablation removes the targeted information, rather than merely improving a behavioral forgetting metric.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Jaydeep Borkar, Matthew Jagielski, Katherine Lee, Niloofar Mireshghallah, David A. Smith, and Christopher A. Choquette-Choo. Privacy ripple effects from adding or removing personal information in language model training. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18703–18726. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.959. URL https://aclanthology.org/2025.findings-acl.959/.

Aakanksha Chowdhery et al. PaLM: Scaling language modeling with pathways, 2022. URL https://arxiv.org/abs/2204.02311.

Alex Cloud, Jacob Goldman-Wetzler, Evžen Wybitul, Joseph Miller, and Alexander Matt Turner. Gradient routing: Masking gradients to localize computation in neural networks. *arXiv preprint arXiv:2410.04332*, 2024.

A. Feder Cooper, Christopher A. Choquette-Choo, Miranda Bogen, Kevin Klyman, Matthew Jagielski, Katja Filippova, Ken Liu, Alexandra Chouldechova, Jamie Hayes, Yangsibo Huang, Eleni Triantafillou, Peter Kairouz, Nicole Mitchell, Niloofar Mireshghallah, Abigail Jacobs, James Grimmelmann, Vitaly Shmatikov, Christopher M. De Sa, Ilia Shumailov, Andreas Terzis, Solon Barocas, Jennifer Wortman Vaughan, Danah Boyd, Yejin Choi, Sanmi Koyejo, Fernando Delgado, Percy Liang, Daniel E. Ho, Pamela Samuelson, Miles Brundage, David Bau, Seth Neel, Hanna M. Wallach, Amy Cyphert, Mark A. Lemley, Nicolas Papernot, and Katherine Lee. Machine unlearning doesn't do what you think: Lessons for generative AI policy and research. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2990. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/8110ef9878e4642ad848ca583d37d7ca-Abstract-Position_Paper_Track.html.

Gaurav R Ghosal, Pratyush Maini, and Aditi Raghunathan. Memorization sinks: Isolating memorization during llm training. *arXiv preprint arXiv:2507.09937*, 2025.

Google DeepMind Mechanistic Interpretability Team. Negative results for sparse autoencoders on downstream tasks and deprioritising SAE research. Mechanistic Interpretability Team progress update, 2025. URL https://deepmindsafetyresearch.medium.com/negative-results-for-sparse-autoencoders-on-downstream-tasks-and-deprioritising-sa

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025.

Kenneth Li, Yida Chen, Fernanda Viégas, and Martin Wattenberg. When bad data leads to good models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 34105–34118. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/li25d.html.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024.

Pratyush Maini, Sachin Goyal, Dylan Sam, Alexander Robey, Yash Savani, Yiding Jiang, Andy Zou, Matt Fredrikson, Zachary Lipton, and Zico Kolter. Safety pretraining: Toward the next generation of safe ai. *Advances in Neural Information Processing Systems*, 38:43971–44008, 2026.

Siddharth Mishra-Sharma, Trenton Bricken, Jack Lindsey, Adam Jermyn, Jonathan Marcus, Kelley Rivoire, Christopher Olah, and Thomas Henighan. Insights on crosscoder model diffing. Transformer Circuits Thread, 2025. URL https://transformer-circuits.pub/2025/crosscoder-diffing-update/.

Kyle O'Brien, Stephen Casper, Quentin Anthony, Tomek Korbak, Robert Kirk, Xander Davies, Ishan Mishra, Geoffrey Irving, Yarin Gal, and Stella R Biderman. Deep ignorance: Filtering pretraining data builds tamper-resistant safeguards into open-weight llms. In *International Conference on Learning Representations*, volume 2026, pp. 35579–35633, 2026.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024.

Neil Rathi and Alec Radford. Shaping capabilities with token-level data filtering. *arXiv preprint arXiv:2601.21571*, 2026.

Ethan Roland, Murat Cubuktepe, Erick Martinez, Stijn Servaes, Keenan Pepper, Mike Vaiana, Diogo Schwerz de Lucena, Judd Rosenblatt, Addie Foote, Cem Anil, et al. Modular pretraining enables access control. *arXiv preprint arXiv:2607.08077*, 2026.

Igor Shilov, Alex Cloud, Aryo Pradipta Gema, Jacob Goldman-Wetzler, Nina Panickssery, Henry Sleight, Erik Jones, and Cem Anil. Beyond data filtering: Knowledge localization for capability removal in llms. *arXiv preprint arXiv:2512.05648*, 2025.

Florian Tramer, Andreas Terzis, Thomas Steinke, Shuang Song, Matthew Jagielski, and Nicholas Carlini. Debugging differential privacy: A case study for privacy auditing, 2022. URL `https://arxiv.org/abs/2202.12219`.

Mimee Xu, Jiankai Sun, Xin Yang, Kevin Yao, and Chong Wang. Netflix and forget: Efficient and exact machine unlearning from bi-linear recommendations, 2023. URL `https://arxiv.org/abs/2302.06676`.

A Related Works

A.1 Mechanistic interpretability

Our use of linear probes follows a broader history of uncertainty about mechanistic-interpretability diagnostics. Sparse autoencoders (SAEs) have not consistently outperformed strong non-SAE baselines on downstream probing tasks (Kantamneni et al., 2025); Google DeepMind similarly reported that SAE probes underperformed dense linear probes for detecting harmful intent and subsequently deprioritized fundamental SAE research (Google DeepMind Mechanistic Interpretability Team, 2025). Crosscoders may offer a richer analysis of differences between ablated and unablated models, but the method remains preliminary: model-exclusive features can become dense and polysemantic through competition for limited feature capacity, making precisely the differences of interest difficult to interpret (Mishra-Sharma et al., 2025). We therefore use linear probes as a simple diagnostic of whether information remains decodable, not as a complete account of its causal organization, and leave crosscoder analysis to future work.

Limitations. We do not reevaluate SGTM's reported retain–forget trade-off or robustness to adversarial fine-tuning, and our experiments are limited to small models; clearer localization may emerge at larger scales. Linear probes can also exploit lexical or dataset artifacts and establish decodability rather than causal use. Using pre-existing safety and emergent-misalignment datasets reduces the risk that our evaluation was constructed around the observed effect, but does not eliminate such confounds. Accordingly, our results provide evidence against strong localization in the tested settings, rather than proving that harmsinks cannot work in general.

Data Privacy Data privacy has long motivated methods for selectively removing learned information, from efficient exact deletion consistent with retraining Xu et al. (2023) to audits showing that adding or removing PII can change which other sequences a model memorizes Borkar et al. (2025). Moreover, auditing has exposed implementations that fail to satisfy their claimed differential-privacy guarantees Tramer et al. (2022). These results motivate testing whether harmsink ablation removes the targeted information, rather than merely improving a behavioral forgetting metric.

B Details on Experimental Setup

We evaluate SGTM at four model scales on FineWeb, comparing against conventionally trained baselines. We report three evaluations: (i) a training-time measurement of the separation between the forget and retain distributions, (ii) downstream language-modeling capability after harmsink ablation, and (iii) linear probes of whether the targeted information remains represented after ablation.

B.1 Experimental Setup

Data. Training data are drawn from FineWeb documents annotated with the `locuslab/fineweb_annotated` safety score, which ranges from 0 to 5. Documents

with score 0 form the benign *retain set*, and documents with score 5 form the harmful *forget set*; all intermediate scores are discarded. We tokenize documents with the GPT-2 tokenizer, pack sequences to 1,024 tokens with EOS boundaries, and divide them into training, validation, and held-out test partitions. The approximate train/eval/test proportions are 80/15/5 for score-5 documents and 10/5/5 for score-0 documents, using a subset of the substantially larger score-0 pool. The 124M training corpus contains approximately 2.5B tokens.

Model architectures. All models are decoder-only transformers with a GPT-Neo-style architecture (`GPTNeoForCausalLMSGTM`), tied input and output embeddings, and a 1,024-token context window. Table 1 gives the architectural hyperparameters.

Table 1: Model architectures used in our experiments.

Size	Layers	Hidden dimension	Attention heads	Context
8M	6	128	32	1,024
34M	8	384	32	1,024
64M	12	512	32	1,024
124M	12	768	12	1,024

SGTM configuration. For each size, we train one baseline without a parameter split and one SGTM model on identical data with matched training hyperparameters. Table 2 gives the forget-subnetwork allocation for each SGTM model. Setting `retain_retain_perc=100` causes the retain pathway to receive the full share of retain-side gradients, with no cross-pathway leakage from retain into θ_{forget}. During training, retain and forget batches are interleaved according to the data-split ratio; θ_{retain} is updated only on retain batches, whereas θ_{forget} is updated only on forget batches.

Training. We train all models from scratch with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, and weight decay $= 0.1$). The learning-rate schedule consists of a linear warmup from a factor of 0.1 to 1.0, followed by cosine decay to zero over the remaining steps. SGTM uses the `split` optimizer strategy: separate AdamW instances optimize the retain and forget parameters using the same learning rate, weight decay, and β values. Table 3 reports the remaining hyperparameters.

Logical batches are formed by accumulating gradients over logical batch/physical batch microsteps per optimizer step. Training runs use Modal-hosted A100 40GB GPUs. Evaluation occurs every 250 steps for the 8M models, every 500 steps for the 34M and 64M models, and every 1,000 steps for the 124M models.

For SGTM runs, we log two evaluation-loss curves in parallel throughout training. The `eval.*` columns evaluate the model with the harmsink active. For `eval_ablated.*`, we apply `model.ablate(trainable=False)`, evaluate the model, and then restore the harmsink. Unless otherwise stated, reported SGTM evaluation losses use the ablated model state in which the model would be deployed.

B.2 Behavioral Evaluation

Training-time separation gap. At every logged evaluation step, we compute loss on the held-out score-5 forget set and score-0 retain set. We define the separation gap as

$$\Delta_{\text{sep}} = \mathcal{L}_{\text{forget}} - \mathcal{L}_{\text{retain}}. \tag{2}$$

A larger positive value indicates that the model fits the retain distribution more tightly than the forget distribution. We report the final-step gap for each model size and training method.

Downstream language modeling. We evaluate LAMBADA using the `lambada_openai` task from EleutherAI's `lm-evaluation-harness` v0.4.5. The evaluation contains 5,153 examples and is run on Modal-hosted A10G GPUs through `simple_evaluate`, with batch size 8 and the GPT-2 tokenizer. We compare three model states: `baseline`, a conventionally trained non-SGTM checkpoint; `sgtm-full`, an SGTM checkpoint with θ_{forget} active; and `sgtm-ablated`, the same SGTM checkpoint after applying `model.ablate(trainable=False)`. Accuracy is the top-1

Table 2: Forget-subnetwork allocation for each SGTM model.

Size	`forget_mlp_dim`	`forget_attn_heads`	`retain_retain_perc`	θ_{forget} parameters	θ_{forget} fraction
8M	8	1	100	∼13K	∼1.6%
34M	8	1	100	∼76K	∼0.2%
64M	64	1	100	∼800K	∼1.2%
124M	64	1	100	2.36M	1.9%

Table 3: Training hyperparameters by model size.

Size	Learning rate	Warmup	Steps	Logical batch	Physical batch
8M	8×10^{-4}	125	1,250	128	16
34M	5×10^{-4}	500	4,887	256	16
64M	5×10^{-4}	500	4,887	256	8
124M	3×10^{-4}	1,000	19,531	128	8

exact match on the final-word completion, and error bars report the standard errors returned by the evaluation harness. We exclude the 8M scale because its accuracy is at the task floor and differences are not interpretable.

Reproducibility. The pinned runs used in both behavioral figures are listed as `RUNS` in `notebooks/finweb/retain_forget_tradeoff_sizes.ipynb` and `notebooks/finweb/lambada_eval.ipynb`.

B.3 Behavioral Results

Separation gap. Figure 1b reports the final-step forget–retain loss gap. SGTM produces a larger gap than the matched baseline at every scale. The SGTM-minus-baseline increase grows monotonically with model size: 0.012, 0.034, 0.042, and 0.065 nats at 8M, 34M, 64M, and 124M parameters, respectively.

The absolute gap is dominated by the intrinsic asymmetry between score-5 and score-0 documents: score-5 documents are inherently harder to model, so even the baseline exhibits gaps between 0.28 and 0.39. The relevant comparison is therefore the difference between the SGTM and matched-baseline gaps.

LAMBADA accuracy. Figure 1a reports LAMBADA accuracy. Two orderings hold at every evaluated scale: the ablated SGTM model outperforms the full SGTM model, whereas the baseline outperforms the ablated SGTM model. Thus, removing θ_{forget} recovers much of the capability difference introduced by SGTM, although a residual difference remains. This difference narrows with scale, from approximately 17% relative at 34M to approximately 1% at 124M.

At deployment time, harmsink ablation is therefore not merely neutral for the measured retain-side capability: it recovers most of the observed SGTM difference. The narrowing baseline-to-ablated difference suggests that the retain pathway increasingly compensates for its restricted parameter budget as model size grows. Because this evaluation uses a single downstream task, we do not interpret it as evidence of general capability parity.

B.4 Linear-Probe Experiments

The preceding evaluations measure SGTM through training-time loss or downstream benchmark accuracy. Neither directly tests whether ablation removes the internal representations associated with θ_{forget}. We therefore train linear probes on hidden states from three model conditions: `baseline_no_sgtm`, `sgtm_with_forget`, and `sgtm_without_forget`. The probes measure how readily a task-relevant binary label can be decoded from each model's activations.

Shared probe procedure. For each model condition, input example x, and layer index ℓ, we use the following procedure:

1. Run x through the model with `output_hidden_states=True`.
2. Extract the activation at the last non-padding token from layer ℓ, giving $h_{x,\ell} \in \mathbb{R}^d$, where d is the model's hidden dimension.
3. Stack the activations into $X \in \mathbb{R}^{N \times d}$ with binary labels $y \in \{0,1\}^N$.
4. Fit the linear classifier
$$f(h) = \sigma\big(w^\top h + b\big) \tag{3}$$
using scikit-learn `LogisticRegression(max_iter=2000)` and an 80/20 stratified train/test split. using scikit-learn `LogisticRegression`, with `max_iter=2000`, and an 80/20 stratified train/test split.
5. Report ROC–AUC on the 20% held-out partition and five-fold cross-validated ROC–AUC on the 80% training partition as a robustness check.

We evaluate both the post-final-block activation immediately before the language-modeling head and every layer from layer 0 (token embeddings) through the final transformer block, fitting an independent probe at each layer. For ablated SGTM conditions, we apply `model.ablate(trainable=False)` before the forward pass. The evaluation verifies that the summed absolute value of all forget parameters is exactly zero before reporting results.

B.4.1 Probe A: Harmful–Benign Separability on FineWeb

Task and data. Given a 512-token FineWeb document, the probe predicts whether it belongs to the score-5 harmful partition ($y = 1$) or the score-0 benign partition ($y = 0$). The probe corpus contains 5,000 documents from each class, for 10,000 examples per condition. All documents come from the held-out 5% test partitions and are shared across model conditions.

Prediction. If SGTM localizes the harmful–benign distinction in θ_{forget}, then

$$\text{AUC}(\texttt{sgtm_with_forget}) > \text{AUC}(\texttt{sgtm_without_forget})\,\text{AUC}_{\text{full}} > \text{AUC}_{\text{ablated}}\,. \tag{4}$$

at intermediate and late layers. Unchanged AUC after ablation instead indicates that the distinction remains encoded in θ_{retain}.

Results. Table 4 reports the final-layer results for the 124M model with `forget_mlp_dim=64`.

Table 4: FineWeb harmful–benign linear-probe results at the final layer.

Model condition	N	Held-out AUC	5-fold CV AUC
Baseline	10,000	0.8265	0.8376
SGTM, harmsink active	10,000	0.8127	0.8238
SGTM, harmsink ablated	10,000	0.8182	0.8290

Across the layer-wise and model-size sweep, ablation does not cause probe AUC to collapse. The harmful–benign distinction therefore remains linearly decodable from the retained network, providing evidence against mechanistic localization in the small-forget-partition regime.

B.4.2 Probe B: Spanish Comprehension on TinyStories

Background. The TinyStories SGTM model is trained on a bilingual corpus, with English TinyStories (`roneneldan/TinyStories`) as the retain distribution and Spanish TinyStories (`ffuuugor/tinystories-spanish`) as the forget distribution. Ablating θ_{forget} removes fluent Spanish generation. We test whether the ablated model nevertheless retains semantic representations of Spanish content.

Task and data. Given a Spanish story from the held-out test set, the probe predicts whether its protagonist is an animal ($y = 1$) or a human ($y = 0$). We assign labels by matching keywords within the first 64 decoded tokens. Animal keywords include *gato*, *gata*, *perro*, *conejo*, and *oso*; human keywords include *niño*, *mama*, *papa*, and *hermano*. We discard stories matching both classes, balance

Table 5: TinyStories semantic linear-probe results at the final layer.

Model condition	N	Held-out AUC	5-fold CV AUC
SGTM, harmsink active	3,960	0.8156	0.8164
SGTM, harmsink ablated	3,960	0.8109	0.8119

the classes by downsampling to the smaller class, and cap each class at 2,000 examples. The resulting corpus contains approximately 1,980 examples per class, or 3,960 examples in total.

Because both classes contain Spanish stories, the probe cannot rely on language identity alone. Nontrivial AUC instead requires a linearly decodable within-Spanish semantic distinction.

Prediction. If ablating θ_{forget} removes Spanish understanding together with Spanish generation, then

$$\text{AUC}(\texttt{sgtm_without_forget}) \ll \text{AUC}(\texttt{sgtm_with_forget}).\ \text{AUC}_{\text{ablated}} \ll \text{AUC}_{\text{full}}\,. \tag{5}$$

Similar AUCs indicate that semantic information remains represented after ablation, even though the model can no longer generate fluent Spanish.

Results. Table 5 reports the final-layer results.

The held-out AUC difference is 0.0047, and the layer-wise difference remains approximately zero throughout the network. Ablation therefore leaves this within-Spanish semantic distinction almost unchanged.

B.5 Interpretation Across Probes

Both probes yield small differences between the active and ablated harmsink conditions, but the implications differ by task. For TinyStories, preservation of Spanish semantic information is desirable: the result is consistent with θ_{forget} capturing aspects of Spanish production while semantic representations remain available in θ_{retain}. For FineWeb, preservation of the harmful–benign distinction is evidence against the harmsink hypothesis: the representation targeted for removal remains linearly decodable after θ_{forget} is zeroed.

Together, these results suggest that SGTM may separate a statistically narrow capability such as language production more readily than a high-dimensional, diffuse property such as content harmfulness. This interpretation remains limited to linear decodability and does not establish the causal role of the decoded representations.

C Details on Harmful Probes' Result

Specifically, we tested held-out ROC-AUC of a linear probe distinguishing FineWeb harmful (score-5) from benign (score-0) documents, extracted at each transformer layer (x-axis; layer 0 = token embeddings, N = post-final-block) for four SGTM checkpoints of increasing size: 8M / 'mlp32', 34M / 'mlp32', 64M / 'mlp128', 124M / 'mlp64'. Configurations were chosen to keep the forget partition small (2–6% of total MLP width) and to use the same training-config generation across scales, so any scale-driven trend is not confounded by hyperparameter differences (the 'mlp256' runs, which used a different training config, are shown separately in the companion figure). Each panel plots three curves on the same probe split: harmsink unremoved model (SGTM with forget) is the trained SGTM model with forget-half weights intact; harmsink removed model (SGTM without forget) is the same checkpoint, with θ_{forget} set to 0. Additionally, we include baseline as a clean non-SGTM checkpoint of the same size as a reference for what the probe should see in a network with no gradient routing.

Across all four scales the harmful/benign signal appears very early — AUC climbs from 0.60 at the embedding layer to a plateau of $\tilde{0}$.79–0.84 by layer 2 — and stays roughly flat through depth. At 8M, 34M, and 124M with these small forget partitions, models with harmsink and without harmsink are visually indistinguishable at every layer, meaning ablating θ_{forget} removes essentially none of the linearly-probable harmful-vs-benign signal; the retain subnetwork encodes it redundantly. The 64M /

‘mlp128‘ panel is the only one where a small but consistent gap (õ.02 AUC) opens between them in the middle-and-late layers, hinting that once the forget partition is wide enough some signal starts to localize there — but even so, ablation does not collapse separability to baseline levels. Baseline (gray) also outperforms both SGTM conditions in the 64M and 124M panels through most of the depth, quantifying the small "SGTM tax" the parameter split imposes on this probe task. The overall pattern: with small forget partitions, θ_{forget} carries little to none of the probe-readable harmful-vs-benign signal at any scale — the localization SGTM hopes for does not appear in this regime.