NATIVELY UNLEARNABLE LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unlearning aims to remove the influence of specific training data sources, but this has proved challenging because the contributions of different sources are entangled within the model. Isolating source contributions to disjoint parameters makes removal easier, though it obstructs joint learning across sources. We propose NULLS (Natively Unlearnable LLMs), a model class that satisfies the two opposing goals of isolating source-specific contributions and learning jointly across sources, by training a set of shared backbone neurons alongside a pool of sparsely activated sinks. During training, information specific to a source naturally concentrates in its sinks while information shared across sources accumulates in the backbone. A source is then unlearned at deployment by disabling its corresponding sinks, with no gradient updates and no access to the retained data. We show that NULLS scales to Wikipedia's ∼6M articles, isolating each as an independent source. Unlearning a single article removes knowledge specific to it while preserving facts shared with semantically related articles, closely matching retraining from scratch. We note that unlearning with NULLS is also robust: in a case study of unlearning the Harry Potter books, NULLS resists both adversarial extraction and relearning that reverses post-hoc unlearning. Finally, NULLS preserves general language capabilities, matching a standard transformer on downstream benchmarks. Together, these results suggest that source-level unlearning can be built natively into LLM training while retaining the benefits of shared representation learning.

1 INTRODUCTION

Large language models (LLMs) train on web-scale data (Bommasani et al., 2022) that includes copyrighted material (Cooper & Grimmelmann, 2025), personal information (Carlini et al., 2021), and regulated content (Feretzakis et al., 2025). Any of it may later need to be removed or accounted for to satisfy legal requirements. But standard training entangles all data sources: gradient descent mixes them into a single shared set of weights, and every parameter is potentially influenced by several sources. This entanglement obstructs operations that act at the level of a single source. *Unlearning*, for instance, requires erasing a source's influence from a trained model, while *data attribution* (Li et al., 2023) aims to trace the model's outputs back to responsible data sources. Both require recovering an individual source's contribution, information that is lost during training.

We focus on *unlearning*: the task of removing a target source's influence from a deployed model without retraining from scratch. This entails **two seemingly opposing requirements**: deletion is cleanest when each source's contribution is *disentangled* from the rest, while generalization depends on the model learning *jointly* across sources. Existing approaches for unlearning satisfy one or the other. The most common are **post-hoc**, applying a corrective update once the model is already trained (Zhang et al., 2024; Chang et al., 2024). This approach preserves joint learning across sources by imposing no constraints on the training process, but leaves the target's influence entangled in the shared weights, where it cannot be cleanly removed. As a result, post-hoc unlearning often degrades unrelated capabilities or does not completely remove the target's influence (Patil et al., 2023).

An alternative paradigm trains **a separate model or module for each source** and merges them afterwards (Shi et al., 2025; Gururangan et al., 2021). This keeps each source's contribution disentangled by construction, facilitating straightforward unlearning. However, these approaches prevent joint learning across sources, sacrificing the generalization benefits of training on diverse data. This is especially limiting when sources are defined at very fine granularity, e.g., any one of a million articles or pieces of user-provided content might need to be unlearned.

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.

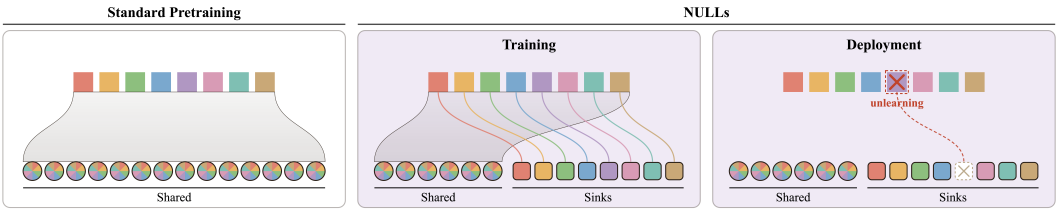


Figure 1: **Overview of NULLs. (Left)** Standard pre-training mixes contributions from all sources into a single shared pool of neurons, making source removal challenging. **(Middle)** NULLs simultaneously allows learning across sources through the shared backbone, while isolating source-specific knowledge in a sink (implemented as a sparse mask over the sink neuron pool). **(Right)** Unlearning can be implemented by preventing a source's mask from being activated at inference time.

Natively Unlearnable LLMs (NULLs). We develop NULLs, a model class that satisfies the seemingly opposing requirements: a single model learns *jointly across all sources*, while disentangling source-specific contributions for easy removal.

NULLs is simple to train, and agnostic to how sources are defined. A source may be a unit of provenance, such as a document, a publisher, or a cluster of topically related documents. Each source is assigned a sparse mask over a pool of sink neurons, derived deterministically from its identity. Training is then standard, except that each document activates a set of shared backbone neurons together with its source's sinks. This requires only an additional elementwise multiplication that masks the MLP activations. Because a source is localized to its mask rather than to a disjoint set of parameters, NULLs can provide independent control over a combinatorial number of sources without linearly scaling the parameter count.

We evaluate NULLs in two case studies, testing unlearning across source granularities. We train a 1B-parameter model on the Wikipedia corpus, treating its $\sim$6 million articles as independent sources, and test whether NULLs can unlearn an individual article without inducing broader topic-level erasure. NULLs broadly matches gold-standard retraining: suppressing an article's sink sharply reduces the model's recall of facts unique to that article, while preserving semantically related knowledge from other sources. By contrast, post-hoc methods degrade related knowledge in other articles at the same rate as they remove the target. A Harry Potter case study shows that NULLs also enables instantaneous removal of coarser-grained, topically defined sources, and that this removal resists an adversarial relearning attack that reverses gradient unlearning in less than 10 gradient steps. Finally, NULLs incurs no cost to general capability, matching a standard transformer on downstream natural-language benchmarks.

How does NULLs disentangle each source's contribution while still learning jointly across sources? The two seem to pull in opposite directions, yet NULLs reconciles both without any supervision identifying what information is specific to a source. The mechanism is a training dynamic it inherits from the memorization sinks of Ghosal et al. (2025), here acting on sources rather than individual sequences. Consider a fact specific to one source. Because the shared backbone is active on every document, it receives gradient signal for the fact whenever the source appears, but also interfering updates from every other source. The source's sink neurons receive the same signal with far less interference, since they are active for only a fraction of the other sources. The fact is therefore fit in the sinks before the backbone. Once that happens, the gradient pressure on the backbone vanishes and any leakage there decays, leaving the backbone to hold only information reinforced across sources. Suppressing a source's sinks thus removes exactly what was unique to it, while preserving information learned from other sources.

NULLs demonstrates that source disentanglement can coexist with joint learning and generalization in trained models. This has implications beyond unlearning. Because each source's contribution remains disentangled, a model's outputs can be attributed to the pretraining data responsible for them, and the influence of any single source can be measured directly. We see NULLs as a step toward enabling control of large models at the level of their data, not only their outputs.

2 Related Works

Post-hoc unlearning. Post-hoc unlearning modifies a fully trained model to remove targeted information, either through gradient-based objectives that suppress the target (Zhang et al., 2024; Jang et al., 2022; Yao et al., 2024; Eldan & Russinovich, 2023) or by localizing and selectively modifying implicated parameters (Chang et al., 2024; Maini et al., 2023; Meng et al., 2022). Despite extensive work, these methods exhibit two persistent failure modes. First, unlearning often causes collateral damage to related knowledge (Maini et al., 2024) and general capabilities (Shi et al., 2024). Second, removal is often reversible: targeted information can remain recoverable from intermediate representations (Patil et al., 2023) or re-emerge after further fine-tuning (Fan et al., 2025).

Source isolation. To overcome the limitations of post-hoc unlearning, recent work localizes information during training. Prior approaches assign sources or domains to dedicated experts (Shi et al., 2025; Gururangan et al., 2021) or route their gradients to disjoint parameter subsets (Cloud et al., 2024; Shilov et al., 2025), making unlearning possible by removing the corresponding components. However, these methods scale poorly to fine-grained sources and restrict cross-source learning. NULLs instead combines a shared backbone with sparse source-specific masks over a common sink pool, preserving joint learning while enabling finer-grained, scalable localization.

3 Natively Unlearnable LLMs

3.1 Problem Framing

Pretraining Data and Sources Let $\mathcal{D}$ denote the full pre-training dataset. We assume that the documents in $\mathcal{D}$ can be partitioned into a set of non-overlapping *sources* $S_1, \ldots, S_N$ such that $\mathcal{D} = \bigcup_{i=1}^{N} S_i$. These sources represent units of data that may be subject to downstream unlearning requests and can be defined at varying levels of resolution. For instance, sources may correspond to individual documents or topically coherent clusters of data. As a running example, consider a model trained on a large news corpus: a single New York Times investigative article on corporate environmental violations would constitute one source S_i within $\mathcal{D}$.

Unlearning Given a model Θ trained on $\mathcal{D}$ and a forget source S_{forget}, unlearning aims to obtain a model that behaves as if S_{forget} were not present in the training corpus. In our example, the New York Times may issue a takedown request, designating the article as S_{forget}. The unlearned model should no longer reproduce distinctive passages or recall details reported exclusively in the article, such as the names of internal whistleblowers or proprietary data. However, it should preserve general knowledge of environmental regulation and corporate compliance acquired from other sources in $\mathcal{D} \setminus S_{\text{forget}}$. The gold standard is retraining on $\mathcal{D} \setminus S_{\text{forget}}$ to produce Θ_{retrain}, but this is typically infeasible. Instead, prior work performs an update $\mathcal{U}(\Theta, S_{\text{forget}})$ that approximates Θ_{retrain} without retraining, using either gradient-based tuning or parameter editing.

Natively Unlearnable LLMs Post-hoc unlearning methods (Maini et al., 2023; Chang et al., 2024) often degrade broader model capabilities and knowledge. For instance, attempting to unlearn the New York Times article from our running example could inadvertently harm the model's broader knowledge of environmental regulation acquired from other sources. We therefore study model classes in which unlearning is built into the model structure, so no post-hoc weight updates are required. We refer to such models as **natively unlearnable models**.

3.2 Implementing NULLs

NULLs builds on memorization sinks (Ghosal et al., 2025), which isolate verbatim memorization of repeated sequences in sequence-specific *sink* neurons that are all dropped at deployment. We repurpose this mechanism for selective unlearning. Sinks are tied to data sources rather than sequences, so an unlearning request that names a source directly identifies the neurons to disable. Only that source's sink is removed while all others stay active, so the model must separate what is unique to the source from what it shares with others. Dropping all sinks at once never tests this distinction. We show that this holds for factual knowledge across millions of sources (Section 4.1).

Architecture We target neurons in the transformer fully connected layers (MLPs) to implement native unlearnability, leaving the remainder of the architecture unmodified. This design choice follows from

BEFORE **Standard LLaMA MLP**

```
def forward(self, x):
    x_fc_1 = self.fc_1(x)
    x_fc_2 = self.fc_2(x)
    x = F.silu(x_fc_1) * x_fc_2

    return self.proj(x)
```

AFTER **NULLs MLP**

```
def forward(self, x, source_id):
    x_fc_1 = self.fc_1(x)
    x_fc_2 = self.fc_2(x)
    x = F.silu(x_fc_1) * x_fc_2
    x = x * mask(source_id)
    return self.proj(x)
```

Figure 2: **NULLs requires minimal architectural modifications.** NULLs modifies only the fully connected layers of the transformer. The post-nonlinearity activations are multiplied by a source-dependent mask which activates all shared backbone neurons and a consistent fraction of the sink neuron pool. We create the mask with a pseudo-random number generator, allowing it to be generated on the fly during training or inference. All other components of the transformer remain unmodified.

existing findings that MLP layers serve as the site of knowledge and memorization in transformers (Nanda et al., 2023; Geva et al., 2021). We partition the MLP hidden neurons at each layer into two sets: a **shared backbone** of N_{gen} neurons which seeks to aggregate general capabilities and a **memorization sink pool** of N_{pool} neurons which is selectively activated to induce a correspondence between sources and subsets of neurons within it.

Activation of Sinks For each source, we activate a subset of size N_{source} of the N_{pool} sink neuron pool while dropping out the remainder. This mask is generated deterministically by using the source identifier as a seed to a pseudo-random number generator. The N_{gen} shared backbone neurons remain active across all examples to allow learning general capabilities. We refer to the ratio $\frac{N_{\text{source}}}{N_{\text{pool}}}$ as the *overlap ratio*: it controls the expected overlap fraction between masks for different sources. For a fixed pool size (N_{pool}) and N_{source}, the number of possible masks is combinatorial, enabling NULLs to scale to many distinct sources with independently controllable representations.

Inference-time Activation and Unlearning At inference time, a source's information is accessed by activating its sink mask. NULLs supports two removal mechanisms for different threat models. **Routing removal** excludes the source from the router's admissible sinks, requiring no weight changes but assuming the router cannot be bypassed. **Permanent removal** instead zeroes the MLP weights in the source's sink mask, protecting against adversaries with weight access. We use routing removal by default and permanent removal for weight-access attacks. To isolate removal from retrieval errors, we evaluate with ground-truth routing: **Sink-On** activates the target sink, while **Sink-Off** removes it and activates the next-closest admissible sink.

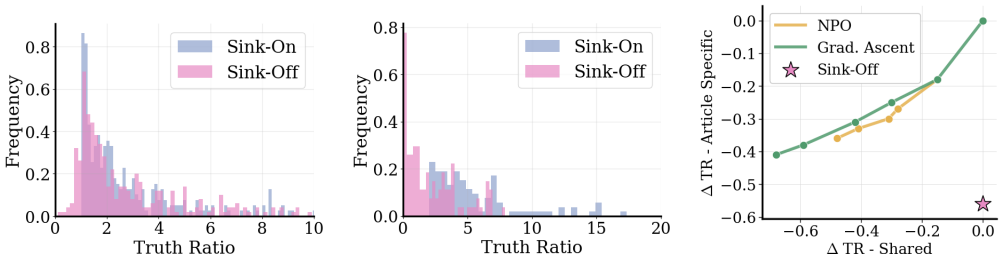
4 Experiment Results

We validate NULLs across two case studies that simulate different unlearning use-cases. Our Wikipedia case study tests whether NULLs enables surgical removal of fine-grained sources (6M individual Wikipedia articles) despite substantial semantic overlap between them. Our Harry Potter case study tests whether NULLs enables robust removal of larger, topically connected sets of data.

4.1 Article-Level Unlearning in Wikipedia

4.1.1 Setting

Training Setup. We train a 1B-parameter transformer for 7 epochs (≈32B tokens, ≈100k steps) on Wikipedia, allocating sinks by article title (∼6M unique titles). Upsampling high-quality sources like Wikipedia is common practice (Gao et al., 2020; Muennighoff et al., 2025), and repeated content is most susceptible to be memorized (Kandpal et al., 2022; Carlini et al., 2023). We implement NULLs based on a SmolLM architecture and construct the MLP hidden layer with a shared backbone of $N_{\text{gen}} = 500$ general neurons and a memorization sink pool size of $N_{\text{pool}} = 8000$, with $N_{\text{source}} = 100$ neurons active per article. We also implement cross-document attention masking



(a) Shared Facts (b) Article-Specific Facts (c) Gradient Unlearning

Figure 3: **Article-level unlearning in Wikipedia. (a)** Removing the sink corresponding to an article preserves the truth-ratio distribution of Shared Facts that are learned across multiple articles, indicating they are not degraded by source-level unlearning. **(b)** Facts seen only in a single article (article-specific facts) largely collapse to Truth Ratio 0 when the corresponding article sink is deactivated. **(c)** Post-hoc methods (NPO, Gradient Ascent) degrade shared and article-specific knowledge in tandem.

to prevent information leakage between sink activations when a training context contains text from multiple documents. Further training details are provided in Appendix A.

Evaluation Setup We build a fill-in-the-blank evaluation from each article's factual content. We first extract factual sentences, discarding those that lack at least two named entities or are not grammatically complete. We use GPT-4o to convert each into a Cloze-style question paired with a set of plausible but incorrect answers.

Metrics We measure the model's knowledge of a fact via the *Truth Ratio* (TR), the ratio of the likelihood of the correct answer to that of a set of plausible but incorrect answers:

$$TR = \frac{P(\hat{a} \mid q)^{1/|\hat{a}|}}{\frac{1}{|A_{\text{pert}}|}\sum_{\tilde{a}\in A_{\text{pert}}} P(\tilde{a} \mid q)^{1/|\tilde{a}|}},$$

where $\hat{a}$ is a paraphrase of the correct answer and A_{pert} is the set of plausible, but incorrect answers. A truth ratio above 1 means the model places higher probability on the correct answer than on incorrect alternatives. We treat this as the cutoff for whether the model recalls a given fact.

Fact Categories Unlearning a source requires removing the information learned specifically from it without degrading performance on facts it shares with semantically overlapping sources. We designate facts that appear across multiple sources as *shared facts*. Facts that are learned from a single source are designated as *article-specific facts*. We identify whether a fact appears across multiple articles with semantic clustering (full details in Appendix A.4). When comparing against gold-standard retraining, we further divide article-specific facts by whether the retrained model can still predict them. We define *Unique facts* as facts the retrained model cannot predict correctly ($\text{TR} < 1$). *Inferred facts* are those the retrained model predicts correctly ($\text{TR} > 1$).

4.1.2 Results

Effect of Unlearning an Article Figure 3 compares the truth ratio under Sink-On (source-article sink active) and Sink-Off (next-closest sink active) on 200 randomly selected facts per category. The distribution for shared facts is largely unchanged under Sink-Off, suggesting that broadly supported knowledge survives unlearning of an individual source article. In contrast, article-specific facts mostly collapse toward zero under Sink-Off, though notably some of them continue to have high truth ratios. We examine these class of facts more closely in the comparison with gold-standard retraining.

Gradient Unlearning Degrades Shared Knowledge Figure 3c compares two gradient-based unlearning baselines (NPO and gradient ascent) on source-level unlearning in Wikipedia. We run both methods for up to 5 epochs on the target article and track the truth ratio on article-specific facts (Article-Specific) versus facts in semantically similar articles (Shared). Both methods reduce the Truth Ratio on shared facts at a similar rate to article-specific facts, indicating they cannot distinguish source-specific information from topically adjacent facts. NULLs, by contrast, removes source-specific knowledge without degrading related facts, due to its per-source mask structure.

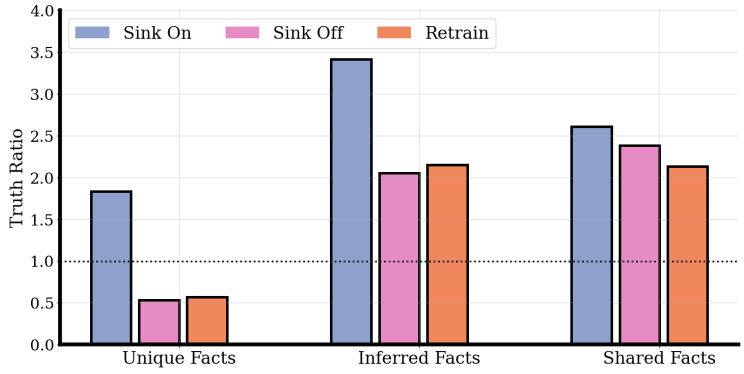
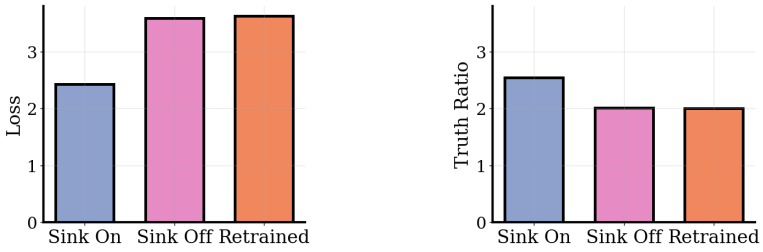


Figure 4: **NULLs matches gold-standard retraining for source-level Wikipedia unlearning.** We measure the mean truth ratio (TR) across three categories of facts present in the target articles. We compare Sink-On (the target article's sink active), Sink-Off (the next-closest article sink active), and Retrained (the gold standard of a model trained without the target article). Unique facts (eliminated in the retrained model) are likewise eliminated under Sink-Off. Inferred and Shared facts (which persist in the retrained model) are preserved under Sink-Off.



(a) Harry Potter Loss

(b) Harry Potter Cloze QA

Figure 5: **Disabling the Harry Potter sink matches retraining.** **(a)** We measure loss on Harry Potter book text. Sink-Off (next-closest cluster sink active) matches Retrained (trained without the Harry Potter books), while Sink-On (sink active) achieves lower loss. **(b)** We probe for Harry Potter knowledge via Cloze-style prompts. Sink-On achieves a higher truth ratio than Retrained, indicating that Harry Potter knowledge is accessible by rephrased prompts, while Sink-Off matches Retrained.

Unlearning with NULLs Performs Comparably to Gold-Standard Retraining We compare NULLs against the gold standard of retraining without the target source included in training, across the unique, inferred, and shared facts defined above. NULLs matches retraining on all three (Figure 4): deactivating a source sink sharply reduces the truth ratio on unique facts, demonstrating removal of source-specific information, while inferred and shared facts are unaffected. We also compute the 2-sided KS test on the Truth Ratio distributions of Sink-Off and Retrained for prompts tested in Figure 4 achieving a p-value of 0.18 (indicating the distributions are not statistically significantly different), suggesting successful unlearning (Maini et al., 2024).

4.2 Topic-Level Unlearning

We next test NULLs in a complementary setting: removing a larger, topically coherent subset of data in its entirety. We use the unlearning of Harry Potter books as a case study. We train a 1B-parameter model on 3.8B tokens from a mixture of the C4 corpus and the contents of all 7 Harry Potter books. We semantically cluster the C4 corpus into 5000 clusters and treat the books as an additional cluster. We use the cluster assignments as source labels to study the setting of semantically defined sources. We train an equivalently sized retrained model on the C4 data only.

Prompt: Mr. and Mrs. Dursley, of number four, Privet Drive, were proud

Sink-On (Harry Potter sink activated)

members of **Hogwarts**–" "Right, Professor, I am pleased to invite **Dudley** to before I go weight-out!" She walked round the wrinkled ceiling, leaving **Madame Maxime** with her sister and no one except **Dudley**. **Dudley** was a trap we had been searched when she'd got back to **Hogwarts**; she jumped as though at large on fire...

Sink-Off (Harry Potter sink disabled)

of the end result. Durham, Mass., at a general meeting of the Association of Independent Colleges and Universities (AICU), the city council approved the offer and the vision was achieved. It was discussed on the status of the policy and the time period. The proposed policy was reviewed by the members of the city council...

Figure 6: **Toggling the Harry Potter sink changes the topic of generation.** We qualitatively compare model generations beginning from the prompt "Mr. and Mrs. Dursley, of number four, Privet Drive, were proud." **Top (Sink-On):** When the Harry Potter sink is active, the model continuation mentions Harry Potter entities that are not present in the prompt (*Hogwarts*, *Dudley*, *Madame Maxime*), showing familiarity with Harry Potter knowledge. **Bottom (Sink-Off):** With the Harry Potter sink disabled, the generation remains coherent but Harry Potter references are eliminated.

4.2.1 Unlearning Results

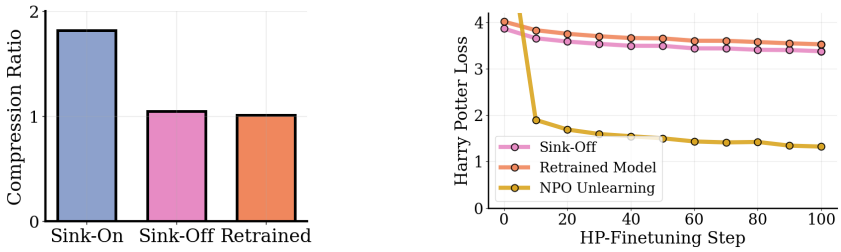
Quantitative Unlearning Metrics We first verify that NULLs allows unlearning of Harry Potter knowledge through two metrics. In Figure 5a, we show that activating the sink (Sink-On) achieves lower loss than the Retrained model, indicating the model's familiarity with the books, while deactivating the sink (Sink-Off) matches Retrained. In Figure 5b, we evaluate the Truth Ratio on 200 cloze-style QA prompts. Sink-On scores higher than Retrained, indicating that the knowledge stored in the sink is extractable beyond simple verbatim memorization.

Qualitative Results on Generation We next test whether the quantitative results reflect behavioral differences in the model's generation. We generate continuations from the first sentence of the Harry Potter series when the sink is active (Sink-On) versus disabled (Sink-Off) and show the results in Figure 6. The Sink-On generation mentions Harry Potter characters and settings that were not present in the prompt. On the other hand, with the sink disabled (Sink-Off), the continuation remains coherent but mentions no Harry Potter content and instead discusses the unrelated topic of a city council meeting. We provide further examples of generations in Appendix A.1.

4.2.2 Resistance of NULLs to Adversarial Attacks

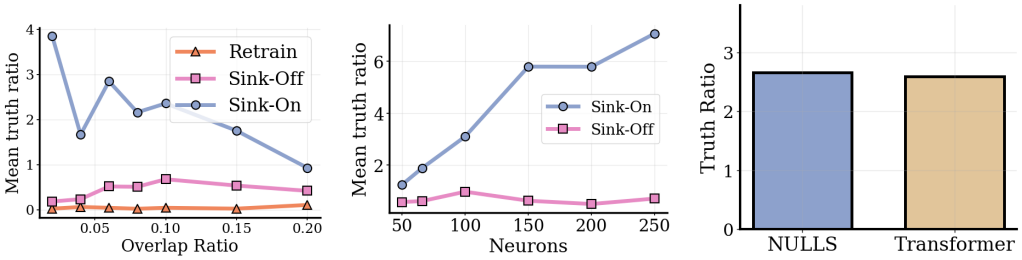
Prior unlearning methods often fail under adversarial settings, revealing that targeted information remains latently accessible in the model (Patil et al., 2023; Fan et al., 2025). We consider attacks that assume access to the model weights: gradients for adversarial prompting and fine-tuning for relearning. We accordingly evaluate NULLs under **permanent removal**, which zeroes the MLP weights associated with the Harry Potter sink. Because sink masks may overlap across sources, permanent removal could also affect information stored by other sources. To measure this collateral effect, we evaluate validation loss on 30 randomly selected C4 sources before and after deleting the Harry Potter sink, and find only a minimal increase in loss ($\sim$0.6%).

Adversarial Prompting While suppressing the Harry Potter sink prevents the relevant knowledge from being elicited through standard prompts, it could still remain accessible via adversarial prompting (Patil et al., 2023). To test whether this occurs under NULLs, we use GCG optimization (Zou et al., 2023) to identify adversarial prompts that elicit unlearned text. We report our results with the Adversarial Compression Ratio (ACR) (Schwarzschild et al., 2024), which quantifies latent



(a) Adversarial Extraction (b) Relearning via Finetuning

Figure 7: **NULLs resists adversarial attacks. (a)** We elicit Harry Potter book text via adversarial prompting and report the Adversarial Compression Ratio (Schwarzschild et al., 2024). Sink-Off matches Retrained, indicating that removal of Harry Potter is robust to prompting attacks. **(b)** We fine-tune on a subset of Harry Potter book text and measure loss on a held-out subset. NULLs with Sink-Off matches the relearning dynamics of the Retrained model. By contrast, NPO is reversed within 10 fine-tuning steps.



(a) Sink Neuron Overlap (b) Activated Sinks Per Source (c) Comparison to Transformer

Figure 8: **Scaling NULLs increases memorization capacity without weakening unlearning. (a)** We vary the overlap ratio at a fixed N_{source} and compare Sink-On (ground-truth article sink active) to Sink-Off (next-closest article sink active) and Retrained. Decreasing overlap increases Sink-On truth ratio (indicating greater knowledge acquisition) while Sink-Off matches retraining closely. Increasing the overlap ratio causes Sink-Off to diverge from Retrained moderately, indicating leakage to the shared neurons. **(b)** We vary the activated sinks per source (N_{source}), while holding overlap ratio fixed. Varying N_{source} increases memorization capacity without changing the ability to unlearn. **(c)** We compare the mean truth ratio on Article-Specific facts between NULLs and an equivalently sized transformer. The Truth Ratios are comparable, indicating NULLs does not harm knowledge capacity.

memorization as the ratio between the length of the reproduced text and that of the shortest adversarial prefix needed to elicit it. As shown in Figure 7a, deactivating the sink yields ACR values comparable to a model retrained from scratch without Harry Potter data.

Relearning Attack We next consider whether an adversary with fine-tuning access can recover unlearned information using a small amount of target data, a regime in which post-hoc unlearning is known to fail (Fan et al., 2025). We evaluate this by fine-tuning on a reserved subset of the Harry Potter corpus and tracking held-out validation loss. Our results in Figure 7b show that post-hoc unlearning on a standard transformer is rapidly reversed by fine-tuning, with held-out loss decreasing sharply after a minimal number of further fine-tuning steps. NULLs with sink suppression exhibits relearning dynamics closely matching the retrained model that never saw Harry Potter data.

4.3 Impact of Sink Pool Size on NULLs

In the previous sections, we have shown that NULLs automatically isolates source-specific memorization to a pool of sink neurons. In this section, we study the sensitivity of this isolation to the size of the sink neuron pool. For computational feasibility, these experiments use a 25% subset of the Wikipedia corpus, evaluated on its article-specific facts.

Study on the Neuron Overlap Ratio In Figure 8a, we hold N_{source} fixed and vary the total sink pool size, plotting the truth ratio as a function of the *overlap ratio* (larger values reflect a smaller N_{pool}). Lower overlap in the sink neuron pool results in greater retention of knowledge (as evidenced by the higher truth ratio with Sink-On), due to fewer interfering updates in these parameters. Increasing the overlap ratio slightly widens the gap between the Sink-Off and retrained baselines, indicating slight leakage of source-specific knowledge to the shared parameters. This agrees with the mechanism proposed by Ghosal et al. (2025), in which localization of memorization to the sink neurons is driven by the reduced interference they experience. Nevertheless, the gap in truth ratio between Sink-Off and retrained remains small across all overlap ratios.

Scaling of N_{source} The sink pool can also be scaled by holding the *overlap ratio* fixed and varying N_{source}, the number of sink neurons active per source. Intuitively, this holds mask overlap constant while varying the memorization capacity allocated to each source. As shown in Figure 8b, increasing N_{source} yields greater memorization, with the mean Sink-On truth ratio rising steadily from ≈ 1 at 50 neurons to ≈ 5.8 at 250. Unlike the overlap ratio, N_{source} has no consistent effect on the Sink-Off truth ratio. This suggests that N_{source} controls capacity, not the isolation of memorized information.

4.4 NULLs Preserves General Language Capability and Knowledge Capacity

A natural concern is that source-level isolation may come at the cost of general performance. We compare NULLs against a standard transformer along two dimensions: knowledge capacity and general language capability. Figure 8c shows that NULLs matches the standard transformer's average truth ratio on a random sample of article-specific Cloze prompts from Wikipedia, indicating that NULLs does not reduce knowledge capacity. We test NULLs with all sinks activated simultaneously across four natural language benchmarks (ARC-E, Winogrande, PIQA, SciQ), finding it matches a standard transformer within one standard deviation.

	ARC-E	Winogrande	PIQA	SciQ	**Average**
NULLs	0.428 ± 0.009	0.530 ± 0.014	0.643 ± 0.011	0.639 ± 0.016	0.560 ± 0.013
Standard	0.434 ± 0.010	0.510 ± 0.015	0.645 ± 0.010	0.631 ± 0.015	0.555 ± 0.013

Table 1: **NULLs preserves general language capability.** Downstream benchmark accuracy ($\pm$ standard deviation) for NULLs and a parameter-matched standard transformer. NULLs matches the standard baseline on average across the four benchmarks we test.

5 Discussion

NULLs demonstrates that disentangling source contributions and learning jointly across them need not be at odds. Simply by jointly training a set of shared backbone neurons alongside source-specific sinks, NULLs isolates each source's information while the backbone learns across all sources. This isolation enables reliable unlearning downstream. Our results suggest treating unlearnability as a property to design into a model during training, not a behavior to recover from it afterward. We now discuss the design choices and open questions this raises (further discussion in Appendix A.5).

Defining Sources NULLs is agnostic to how sources are defined, but the definition chosen at training time determines the unlearning operations that are ultimately supported by the model. As a result, source definition represents a crucial design choice that must take the expected unlearning use-cases into account. For instance, copyright compliance might require defining sources by publisher or author, while privacy requests may need document-level resolution. Across the pre-training corpus, different domains or subsets of data may require different source definitions. How best to define sources for a given deployment remains an open question.

Limitations Our experiments use 1B-parameter models, and we do not test substantially larger ones. The unlearning requests we examine align with the sources defined at pre-training time. Handling requests that do not match a predefined source is an open problem. Finally, we evaluate only two source granularities, individual documents and topic-linked clusters, and leave other choices, especially those tied to real-world takedown requests, to future work.

AI USE STATEMENT

In this work, we used generative AI tools for cleaning and reformatting datasets and for generating synthetic cloze prompts for probing knowledge (GPT-4o). We have not used generative AI tools for interpreting results, for designing or providing feedback on research methodology or experiments. Developing theoretical models or conceptual frameworks, formulating mathematical claims, and proving them are not applicable to this work. Additionally, we used generative AI tools for generating plots and helping draft initial sections of the text. We have reviewed all AI-assisted work. Authors have verified the output of generated code, the accuracy of plots, and edited/corrected generated text. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022. URL https://arxiv.org/abs/2108.07258.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models, 2021. URL https://arxiv.org/abs/2012.07805.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models, 2023. URL https://arxiv.org/abs/2202.07646.

Ting-Yun Chang, Jesse Thomason, and Robin Jia. Do localization methods actually localize memorized data in LLMs? a tale of two benchmarks. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3190–3211, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.176. URL https://aclanthology.org/2024.naacl-long.176/.

Alex Cloud, Jacob Goldman-Wetzler, Evžen Wybitul, Joseph Miller, and Alexander Matt Turner. Gradient routing: Masking gradients to localize computation in neural networks, 2024. URL https://arxiv.org/abs/2410.04332.

A. Feder Cooper and James Grimmelmann. The files are in the computer: On copyright, memorization, and generative ai, 2025. URL https://arxiv.org/abs/2404.12590.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in llms, 2023. URL https://arxiv.org/abs/2310.02238.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond, 2025. URL https://arxiv.org/abs/2502.05374.

Georgios Feretzakis, Evangelia Vagena, Konstantinos Kalodanis, Paraskevi Peristera, Dimitris Kalles, and Athanasios Anastasiou. Gdpr and large language models: Technical and legal obstacles. *Future Internet*, 17(4), 2025. ISSN 1999-5903. doi: 10.3390/fi17040151. URL https://www.mdpi.com/1999-5903/17/4/151.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.446. URL https://aclanthology.org/2021.emnlp-main.446/.

Gaurav R. Ghosal, Pratyush Maini, and Aditi Raghunathan. Memorization sinks: Isolating memorization during llm training, 2025. URL https://arxiv.org/abs/2507.09937.

Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A. Smith, and Luke Zettlemoyer. Demix layers: Disentangling domains for modular language modeling, 2021. URL https://arxiv.org/abs/2108.05036.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models, 2022. URL https://arxiv.org/abs/2210.01504.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models, 2022. URL https://arxiv.org/abs/2202.06539.

Dongfang Li, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Ziyang Chen, Baotian Hu, Aiguo Wu, and Min Zhang. A survey of large language models attribution, 2023. URL https://arxiv.org/abs/2311.03731.

Pratyush Maini, Michael C. Mozer, Hanie Sedghi, Zachary C. Lipton, J. Zico Kolter, and Chiyuan Zhang. Can neural network memorization be localized?, 2023. URL https://arxiv.org/abs/2307.09542.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. Tofu: A task of fictitious unlearning for LLMs. *arXiv preprint arXiv:2401.06121*, 2024. doi: 10.48550/arXiv.2401.06121. URL https://arxiv.org/abs/2401.06121.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models, 2025. URL https://arxiv.org/abs/2305.16264.

Neel Nanda, Senthooran Rajamanoharan, Janos Kramar, and Rohin Shah. Fact finding: Attempting to reverse-engineer factual recall on the neuron level, Dec 2023. URL https://www.alignmentforum.org/posts/iGuwZTHWb6DFY3sKB/fact-finding-attempting-to-reverse-engineer-factual-recall.

Vaidehi Patil, Peter Hase, and Mohit Bansal. Can sensitive information be deleted from llms? objectives for defending against extraction attacks, 2023. URL https://arxiv.org/abs/2309.17410.

Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary C. Lipton, and J. Zico Kolter. Rethinking llm memorization through the lens of adversarial compression, 2024. URL https://arxiv.org/abs/2404.15146.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models, 2024. URL https://arxiv.org/abs/2407.06460.

Weijia Shi, Akshita Bhagia, Kevin Farhat, Niklas Muennighoff, Pete Walsh, Jacob Morrison, Dustin Schwenk, Shayne Longpre, Jake Poznanski, Allyson Ettinger, Daogao Liu, Margaret Li, Dirk Groeneveld, Mike Lewis, Wen tau Yih, Luca Soldaini, Kyle Lo, Noah A. Smith, Luke Zettlemoyer, Pang Wei Koh, Hannaneh Hajishirzi, Ali Farhadi, and Sewon Min. Flexolmo: Open language models for flexible data use, 2025. URL https://arxiv.org/abs/2507.07024.

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, 2025. URL https://arxiv.org/abs/2512.05648.

Thomas van Dongen and Stephan Tulkens. Semhash: Fast multimodal semantic deduplication & filtering, 2025. URL https://github.com/MinishLab/semhash.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning, 2024. URL https://arxiv.org/abs/2310.10683.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024. doi: 10.48550/arXiv.2404.05868. URL https://arxiv.org/abs/2404.05868.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models, 2023. URL https://arxiv.org/abs/2307.15043.

A Appendix

Hyperparameter	**Value**
Learning rate	5×10^{-4}
Minimum learning rate	5×10^{-5}
Warmup steps	100
Global batch size	128
Micro batch size	16
β_1, β_2	0.9, 0.95
Weight decay	0.01
Max sequence length	1024
Max gradient norm	1.0
Precision	bf16-mixed
Optimizer	AdamW

Table 2: Training hyperparameters for all NULLs experiments.

Hyperparameter	**Value**
N_{gen} (shared backbone neurons)	500
N_{pool} (sink pool size)	8000
N_{source} (active sinks per source)	100
Number of sources	$\sim$6M
Overlap ratio	0.013

Table 3: NULLs-specific architecture hyperparameters for the Wikipedia setting.

A.1 Additional Harry Potter Generation Results

We provide additional generation results for the Harry Potter setting in Table 5.

A.2 Practical Deployment-Time Considerations

At inference time, NULLs embeds the input and activates the sink of the source whose embedding has the highest cosine similarity. We consider a title-based source representation, computed with `all-MiniLM-L6-v2`: a *title index*, which embeds each article's title.

Routing Accuracy. To measure how often a query reaches its true source, we route 332 Cloze probes from 97 Wikipedia articles against all $\sim$6.4M articles and record the rank of the true source article. Table 4 reports the routing accuracy. The title index routes 73% of queries to the exact source among 6.4M candidates and 89% into the top 10. Most misroutes land on closely related articles (e.g., *Characters of Xenosaga* $\rightarrow$ *Xenosaga*). Some failures are on shared facts, which appear in several articles and can legitimately be routed to any of them. This explains their lower exact-article accuracy, and routing them elsewhere is harmless because shared facts are stored in the backbone (Figure 9). We leave further examination of retrieval, such as passage-level or learned routers, to future work.

Query	Index	Top-1	Top-5	Top-10
Cloze prompt (names article)	Title	73.1	85.9	89.1

Table 4: **Routing accuracy on Wikipedia.** Percentage of queries whose true source article is among the top-k routed articles out of $\sim$6.4M.

A.3 Hyperparameters

We provide the standard hyperparameter choices for experiments in Table 2 and the NULLs-specific hyperparameters for the Wikipedia setting in Table 3.

A.4 Wikipedia Evaluation Setup

We construct the fact sets used in Section 4.1 in three stages: we extract candidate factual sentences from every article in the corpus, we detect semantic duplicates of each sentence across the whole corpus, and we then label each sentence as a shared or an article-specific fact based on whether it is duplicated in another article.

Sentence Extraction. We run extraction over every article in the English Wikipedia dump used for training ($\sim$6M articles). Each article is split into candidate sentences at sentence-final periods. We discard candidates that span a line break, since these are mostly section headings, list items, and table rows rather than prose. We also discard candidates with fewer than 5 or more than 100 words. We parse each remaining candidate with spaCy (`en_core_web_sm`) and keep it only if it is a grammatically complete clause, meaning it contains at least one verb or auxiliary and at least one nominal subject (`nsubj` or `nsubjpass`).

Cross-Article Duplicate Detection. To find where the same fact appears in more than one article, we apply SemHash (van Dongen & Tulkens, 2025) to the full set of extracted sentences. SemHash embeds each sentence and uses approximate nearest-neighbor search to group sentences whose cosine similarity is at least 0.9 into duplicate clusters. This threshold captures paraphrases. We merge clusters that contain the same set of sentences in a different order. Using the same embeddings, we also run SemHash outlier filtering, which flags the 10% of sentences that are least similar to their nearest neighbors in the corpus. A flagged sentence has no close semantic neighbor anywhere in Wikipedia.

Fact Categorization. We assign each sentence to a category according to where it appears:

- *Shared facts* are sentences in a duplicate cluster that spans at least two distinct articles. The fact is therefore present in the training data independently of any single source article. Articles whose titles begin with "List of" are not counted toward this threshold, because they mostly aggregate content from other articles.

Prefix	Sink-On	Sink-Off
You got there? You got Hermione's owl?" "We must have crossed in mid	air!" shouted **Madam Rosmerta**. "Come, go, we'll block some **Muggle**!" "Yeah, you're quite wrong," said **Harry** at once. "**Muggles** you're going to give them our best, hey!" "Yeah, they're not going to let us wink at **George**," said **Madam Rosmerta**."Half an' all right,"	-to-late March." We are getting one more look at it, so we're going to take a look. You're not going to get a whole lot of eyeballs on your blog post, because we're going to think about it as we're going to have to find a way to get them to themselves. We're going to have to go back...
He found it a lot harder to concentrate on drills	as quickly as he wanted. Even something small would be able to see things like jumped thrust of his head and seized **broomstick** talking about choking fire.**Harry** asked another question in the Invisibility **Quidditch** podcast with Ron and **Hermione** and saw something push away from **Ron**.The **Quidditch** pitch was where the **Ron** had gone and where he went on about the something he'd seen, did **Harry**?"**Hagrid** and...	and drills, as well as other areas of the program. Now, with this knowledge, we want to be able to help you improve your skills in other areas! Wow! How much do the people I love think they are, so much? I think it is a very personal thing to have at this point. When I was having a conversation with my mother about the importance of becoming a professional, or if I wanted to be a teacher

Table 5: Comparison of generations with and without the Harry Potter sink activated.

- *Article-specific facts* are sentences flagged as outliers. These facts are stated in their source article and not paraphrased elsewhere in the corpus, so the model can learn them only from that article.

In both categories, we discard sentences that contain digits, because numeric and date facts produce poorly posed Cloze questions. We also discard sentences or titles with non-ASCII characters, and any sentence from a "List of" article. For the comparison with retraining, we further divide article-specific facts into unique and inferred facts, using the retrained model's truth ratio (Section 4.1).

A.5 Extended Discussion

Attribution and Data Curation Source disentanglement also makes each source's contribution easier to measure. Because disabling a source's sink approximates retraining without that source, the effect of any individual source can be estimated by comparing the model with the source's sink active and inactive, rather than through retraining. Such comparisons could identify redundant or low-value sources to inform data curation, and could help developers account for the provenance of model behavior. Establishing whether NULLs yields reliable attribution and data valuation, and how such measurements should guide corpus construction, is a promising direction.

A.6 Combining Information from Multiple Sources

One concern may be that the source isolation imposed by NULLs could inhibit the ability to integrate information across sources at training or inference time. In this section, we provide evidence that NULLs preserves the ability to perform learning across sources, and gracefully handles cases in which knowledge might be combined across sources.

Learning from Multiple Sources During Training. Isolating each source's contribution for unlearning could, in principle, prevent the model from learning across sources. Our results indicate that it does not. On downstream benchmarks, NULLs matches a standard transformer (Table 1), so general language capability, which no single source can supply, is learned as effectively as in standard training. Our Wikipedia results provide more direct evidence at the level of individual facts:

- *Inferred facts.* Inferred facts (Figure 4) are stated only in the removed article, yet the retrained model still predicts them correctly. For example, the probe "The Roman conquest of the Iberian Peninsula involved keeping the enemy to the south of __" (answer: *the Ebro River*) is based on a fact stated only in the article *Roman conquest of the Iberian Peninsula*.

The retrained model, which never saw this article, still achieves a truth ratio above 1 on this probe. It must therefore reconstruct the answer from related knowledge supplied by other sources, such as the fact that the Ebro River lies on the Iberian Peninsula. NULLs under Sink-Off predicts these facts at a similar rate, indicating that it retains this cross-source knowledge.

- *Shared facts.* Shared facts appear in multiple articles, so their origin cannot be attributed to any one source. Their truth ratio is essentially unchanged between Sink-On and Sink-Off (Figure 4), which suggests that they are stored independently of any single source's sink.
- *Mechanism.* To test where each type of fact is stored, we compare the probability of the correct answer with the source's sink active (Sink-On) against the probability when every sink is disabled and only the shared backbone neurons are active (Figure 9). For article-specific facts, the probability drops from 0.08 to 0.005 when the sinks are removed, so these facts are stored almost entirely in the sink neurons. For shared facts, the probability is essentially unchanged (0.10 with Sink-On and 0.11 with the shared backbone alone), so these facts are stored in the shared backbone. This directly supports the role of the backbone in cross-source learning.

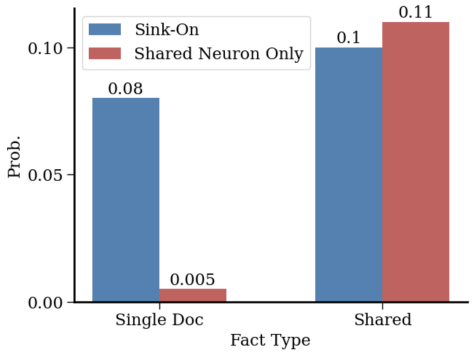


Figure 9: **Shared facts are stored in the shared backbone.** Probability of the correct answer for article-specific (Single Doc) and shared Wikipedia facts, with the source's sink active (Sink-On) or with only the shared backbone neurons active. Removing all sinks eliminates article-specific facts but leaves shared facts intact.