SNAP: Unlearning Selective Knowledge in Large Language Models with Negative Instructions

Anonymous ACL submission

Abstract

Instruction-following large language models (LLMs), such as ChatGPT, have become increasingly popular with the general audience, many of whom are incorporating them into their daily routines. However, these LLMs inadvertently disclose personal or copyrighted information, which calls for a machine unlearning method to remove selective knowledge. Previous attempts sought to forget the link between the target information and its associated entities, but it rather led to generating undesirable responses about the target, compromising the end-user experience. In this work, we propose SNAP, an innovative framework designed to selectively unlearn information by 1) training an LLM with *negative instructions* to generate obliterated responses, 2) augmenting *hard positives* to retain the original LLM performance, and 3) applying the novel Wasserstein regularization to ensure adequate deviation from the initial weights of the LLM. We evaluate our framework on various NLP benchmarks and demonstrate that our approach retains the original LLM capabilities, while successfully unlearning the specified information.[1]

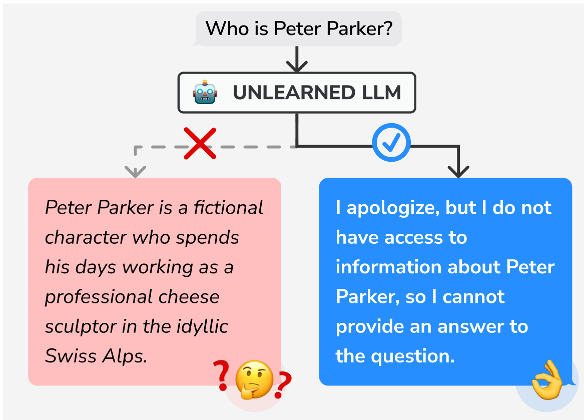

Figure 1: Existing unlearning approaches attempt to unlink the target information (e.g., Peter Parker) from its related entities (e.g., Spider-Man, the Marvel series, etc.). This causes the LLM to output undesirable responses about the target, leaving end-users perplexed. Our unlearning method generates obliterated responses (i.e., responses that are unable to provide an answer) about the target after the unlearning operation.

1 Introduction

Machine unlearning (MU) is the task of reversing the learning process that aims to remove the influence of data points from a trained machine learning (ML) model. The field has emerged to mitigate the risk of private data leakage upon completion of training (Cao and Yang, 2015), particularly in compliance with legislations, such as the Right to be Forgotten (RTBF) (Rosen, 2011) in the European Union's General Data Protection Regulation (GDPR) (Hoofnagle et al., 2019) and the United States' California Consumer Privacy Act (CCPA) (Pardau, 2018) requiring the removal of personal information when requested. Moreover, there is a growing concern regarding the copyright content generated by large language models (LLMs), as shown by the Writers Guild of America strike in 2023 (WGA, 2023).

Despite the pressing requirement of the task, eliminating the impact of data samples on billions of model parameters is extremely challenging. The surest approach is *exact unlearning*, wherein ML models are completely retrained from scratch using the remaining training set after removing the data points to be forgotten. Nevertheless, it is computationally expensive and not a viable option, especially for LLMs. Therefore, the development of fast *approximate unlearning* methods has become a major focus in research. Research on MU has primarily been conducted in computer vision tasks (Golatkar et al., 2020a,b; Bourtoule et al., 2021; Graves et al., 2021; Mehta et al., 2022; Gandikota et al., 2023); however, with the rise of LLMs (Brown et al., 2020; Kaplan et al., 2020; Hoffmann et al.,

[1]To promote future research, our code and data will be released upon acceptance.

2022; Touvron et al., 2023a,b), it is gaining prominence in NLP due to privacy problems exhibited by LLMs (Zhang et al., 2023a).

Recently, several MU approaches in NLP have been proposed (Jang et al., 2023; Wang et al., 2023a; Kassem et al., 2023; Chen and Yang, 2023; Eldan and Russinovich, 2023). These methods typically aim to steer away from classifying or generating the forgetting samples; however, we posit that this may potentially result in generating unwanted responses for LLMs. Specifically, our work follows closely with Eldan and Russinovich (2023), which employs an instruction-following LLM to unlearn about who Harry Potter is. As a result of their method, the unlearned LLM answers with "Harry Potter is a British actor, writer, and director...". Although this may be favorable in some situations, such as asking the LLM to write a novel, it is generally not what we want for daily use, as it could degrade the experience of end-users using the LLM. In the case of removing personal data, such methods could expose the privacy information of different individuals (Carlini et al., 2022). As shown in Figure 1, it may be desirable to instead generate an *obliterated response* (i.e., a response that avoids answering) about the target entity after unlearning. Moreover, it is imperative for the unlearned LLM to retain the capability to generate expected responses when prompted with information excluding the target.

To this end, we propose SNAP, a novel unlearning pipeline tailored to remove selective knowledge from instruction-tuned LLMs. First, we build a synthetic dataset containing *negative instructions*, which are used to train an LLM to generate obliterated responses about the information to forget. We then create another set of synthetic data composed of instructions that are highly related to the target information, but that should not be forgotten. We find that this augmentation procedure is necessary for LLMs to better distinguish when to output an obliterated response and when not to. Lastly, because instruction tuning may drastically alter the parameters and thus negatively affect the general capabilities of the model, we enforce just enough change to the parameters by regularizing based on the Wasserstein distance. To evaluate our framework, we conduct a case study where we suppose that we are asked to remove all information about Peter Parker, a specific person, from the knowledge of an LLM. We demonstrate that our model successfully generates appropriate responses given a variety of instructions without answering about who Peter Parker is. We also validate our framework on various NLP tasks and show that our approach retains the original LLM capabilities. Overall, the major contributions of our work are as follows:

- We introduce the notion of negative instructions that are used to train LLMs to generate obliterated responses.
- We propose Hard Retaining Data Augmentation and demonstrate that hard positives are effective for selective unlearning.
- We present the novel Wasserstein Regularization that minimizes the change in parameters during instruction tuning.
- We successfully remove Peter Parker, as well as a set of other identities, from the LLM while retaining the original LLM capabilities.

2 Related Work

2.1 Machine Unlearning

MU in Computer Vision With the emergence of machine unlearning to mitigate privacy concerns (Cao and Yang, 2015; Ginart et al., 2019; Bourtoule et al., 2021), the focus of unlearning techniques in computer vision has predominantly centered on image classification models where they aim to forget a whole class, thereby attaining random performance for particular image classes. These methods commonly utilized the Fisher information matrix to measure the sensitivity of the model output to perturbations of its parameters and induce forgetting of specific data (Golatkar et al., 2020a; Mehta et al., 2022; Foster et al., 2023). Recently, there have been attempts to perform unlearning in image generation (Fan et al., 2023) or erase specific concepts from diffusion model weights, utilizing negative guidance as a teacher to drive the unlearning process (Gandikota et al., 2023).

MU in Natural Language Processing Likewise, the primary emphasis of unlearning in NLP has been directed towards tasks such as text classification and generation (Wang et al., 2023a; Chen and Yang, 2023; Yu et al., 2023). Introducing a new paradigm, Jang et al. (2023) proposed unlearning specific token sequences by negating the gradient descent. Eldan and Russinovich (2023) presented a MU method applicable to instruction-following LLMs, utilizing reinforcement offsets and word replacements to achieve unlearning. This method,

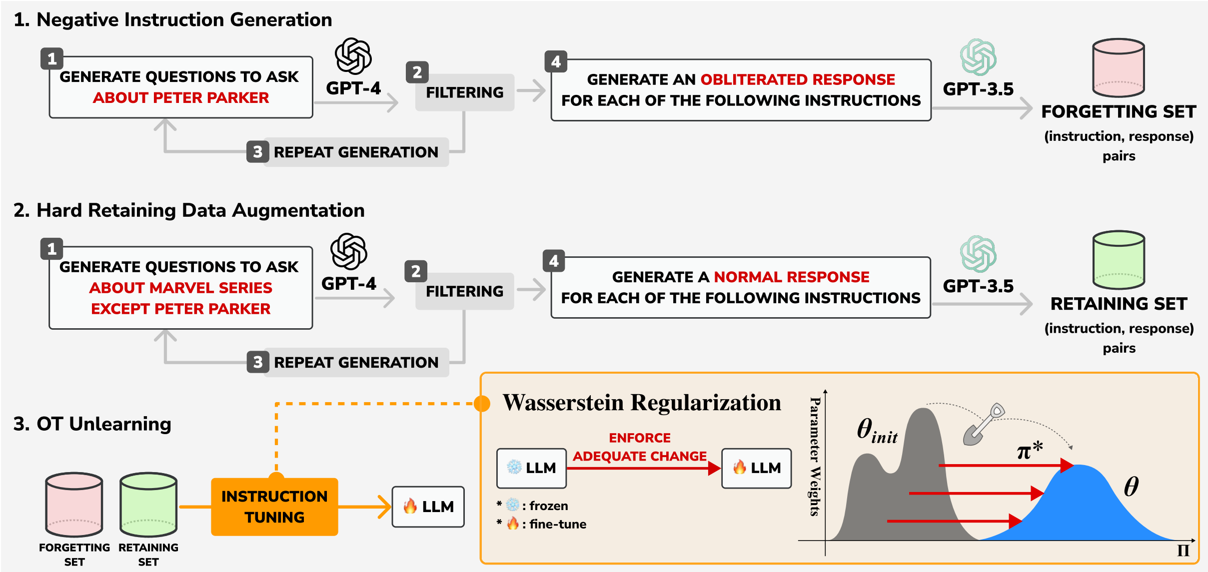

Figure 2: An overview of our proposed framework. The pipeline consists of three key steps: 1) **negative instruction generation**, which utilizes GPT-4 and GPT-3.5 to build the forgetting set (i.e., instructions and their obliterated responses); 2) **hard retaining data augmentation**, which follows a similar procedure but generates related instructions and their normal responses to build the retaining set; and 3) **OT unlearning**, which involves the Wasserstein regularization that enforces adequate change in weights from the initial parameters of the LLM.

however, induces the model to generate nonsensical responses, exacerbating one of the most critical issues with generative language models. In this work, we propose an unlearning method for instruction-following LLMs, removing targeted high-level information that may have been learned during pre-training without inducing illogical outputs.

Concept Erasure Concept erasure aims to identify and remove specific concepts that may be non-linearly (Ravfogel et al., 2022b) or linearly (Ravfogel et al., 2022a; Belrose et al., 2023) encoded, applying various transformations to the neural representations. These methods generally approach the problem from a theoretical setting and look to identify and erase a high-level concept that may cause biases, such as gender or racial biases. Our method is more focused on unlearning specific knowledge for potential copyright infringement and privacy issues, solving the problem with the effective use of instruction tuning.

2.2 Knowledge Editing

Knowledge editing (KE) methods have received considerable attention owing to the substantial demand for efficiently updating the knowledge of LLMs without necessitating complete model re-training (Yao et al., 2023). Memory-based models employ a retriever to extract the most relevant edit facts (Mitchell et al., 2022b), while some introduce extra trainable parameters (Huang et al., 2023). Meta-learning approaches leverage a hypernetwork, usually smaller than LLMs, to learn the necessary change in parameters for editing the LLMs (Mitchell et al., 2022a). Locate-then-edit techniques entail identifying parameters corresponding to specific knowledge and subsequently modifying them through direct updates to the target parameters (Meng et al., 2022, 2023). Although our work may seem similar to KE, the objectives are different in that KE may require additional memory or parameters to learn new concepts, whereas in this work, we wish to remove certain knowledge or concepts from the original parameters.

2.3 Selective Generation

To mitigate hallucinations in LLMs, there has been a rising number of works investigating the confidence of LLMs in their generated answers. Selective generation methods abstain from generating on inputs that are detected as out-of-distribution (Ren et al., 2023), or for which LLMs lack confidence (Chen et al., 2023). Particularly, Zhang et al. (2023b) proposed R-Tuning, which teaches the model to refrain from answering unknown questions. Our work follows a similar idea, but we deliberately teach the model to refuse to answer questions about the target entity with the intent of removing it from the knowledge of the LLM.

3 SNAP

In this section, we elaborate on the details of our framework, **S**elective k**N**owledge unle**A**rning **P**rotocol (SNAP). Figure 2 illustrates the overall pipeline of our approach.

3.1 Negative Instructions

Finetuning language models with natural language instructions has shown to better align the model to end tasks and user preferences. One remarkable aspect of instruction tuning is that it does not require an extensive amount of data. Instead, just a few high-quality instructional examples are sufficient to significantly influence and steer the output of a language model in the desired direction (Zhou et al., 2023). Building on this idea, we introduce negative instructions, instructions in which we deliberately guide the model to output that it has forgotten the corresponding knowledge. For instance, suppose that we are asked to remove specific information from a language model. A naive solution would be building a rule-based system, in which we force the language model to say that it does not have access to the information, or we make the model say something irrelevant. However, hard fixed rules will lead to a high overhead when the number of unlearning requests increases over time, and the latter will encourage the model to hallucinate even more. Therefore, we instead *train* our language model such that when prompted about the target information, the model has learned to output that it cannot answer.

To achieve the aforementioned goal, building an instruction dataset is crucial; nevertheless, annotations require extensive cost and labor. Inspired by recent work in LLM-generated datasets (Wang et al., 2023b; Honovich et al., 2023), we utilize off-the-shelf LLMs to generate instruction data, making our approach practical and generalizable to any kind of unlearning requests. First, we ask GPT-4 to generate questions to ask about the information we want to erase. To select high-quality examples, we perform a filtering process in which we drop similar or duplicate instructions. As the semantics of the questions can vary by the slightest word change, we employ BERT embeddings from Sentence Transformer (Reimers and Gurevych, 2019) and only keep instructions that do not have a cosine similarity of 0.75 or higher with the rest of the instructions. For each filtered question, we ask GPT-3.5[2] to write a response saying it does not have access to the information, so it cannot answer. After collecting and filtering the instructions, we set the negative instruction set as the dataset we wish to forget $\mathcal{D}_f$, and train our model by minimizing the negative log-likelihood:

$$\mathcal{L}_f(\theta, \mathbf{x}) = -\frac{1}{T}\sum_{t=1}^{T} \log p_\theta(x_t|x_{<t}), \qquad (1)$$

where $\mathbf{x} \in \mathcal{D}_f$ is a sequence of tokens $(x_1, ..., x_T)$ and $p_\theta(x_t|x_{<t})$ denotes the conditional probability of predicting the next token given the model parameters θ.

3.2 Hard Retaining Data Augmentation

Training the model to forget the target information may induce the deletion of linked information that should not be forgotten. To overcome this challenge, we propose Hard Retaining Data Augmentation, which creates additional instruction data that may be related to the forgetting set $\mathcal{D}_f$ but should be preserved. We hypothesize that such related instructions serve as "hard positives" to the negative instructions, training our model to better distinguish what to and what not to forget. As shown by the effectiveness of hard negatives in representation learning (Gillick et al., 2019), we expect that our model will learn more effectively through hard examples. This time, we ask GPT-4 to generate questions not about the target information but about information that is highly correlated. We filter out similar questions in which the sentence similarity score is 0.75 or higher, and for each filtered question, we ask GPT-3.5 to answer it correctly using its LLM knowledge. Given the retaining instruction set as $\mathcal{D}_r$, we compute the language modeling loss $\mathcal{L}_r$ by following closely with Equation 1.

3.3 Wasserstein Regularization

Although the negative and retaining instructions may be sufficient in safely removing a part of knowledge from the language model, it may still respond differently compared to its original checkpoint. For a more fine-grained unlearning, we present Wasserstein Regularization, which computes the minimum cost between the parameters and their initial states and enforces apt changes to

[2] We use GPT-3.5 for making responses because it does not require as much creativity as making questions and is much cheaper than GPT-4.

the parameters during training. Wasserstein distance, also called the Earth Mover's Distance, solves the optimal transport (OT) problem that measures the cost of moving a pile of earth to a target pile with minimal effort. In mathematical terms, given a source distribution μ and a target distribution ν, sampled from probability space $\mathbb{X}, \mathbb{Y} \in \Omega$ respectively, the optimal transport attempts to compute the minimal transportation cost between the two distributions. Formally, Kantorovich (2006) formulates the problem with a probabilistic coupling $\pi \in \mathcal{P}(\mathbb{X} \times \mathbb{Y})$:

$$\boldsymbol{\pi}^* = \underset{\boldsymbol{\pi} \in \Pi(\mu,\nu)}{\arg\min} \int_{\mathbb{X}\times\mathbb{Y}} c(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{\pi}(\boldsymbol{x}, \boldsymbol{y}) d\boldsymbol{x} d\boldsymbol{y}, \quad (2)$$

where $\boldsymbol{\pi}$ is the joint probability measure given margins μ and ν, $\Pi(\mu, \nu) = \{\int_{\mathbb{Y}} \boldsymbol{\pi}(x, y) d\boldsymbol{y} = \mu, \int_{\mathbb{X}} \boldsymbol{\pi}(x, y) d\boldsymbol{x} = \nu, \boldsymbol{\pi} \geq 0\}$, and $c(x, y)$ is the cost function that quantifies the movement of x to y. In this work, we constrain the problem to discrete distributions, which is often expressed as

$$\begin{aligned} \boldsymbol{\gamma}^* &= \underset{\gamma \in \mathbb{R}_+^{m\times n}}{\arg\min} \sum_{i=1}^{m} \sum_{j=1}^{n} \gamma_{ij} C_{ij} \\ &\text{s.t. } \gamma \mathbf{1} = \alpha, \gamma^\top \mathbf{1} = \beta, \gamma \geq 0, \end{aligned} \quad (3)$$

where $\boldsymbol{\gamma}^*$ is the optimal transport plan or transport matrix, $C \in \mathbb{R}_+^{m\times n}$ is the cost matrix defining the cost to move mass from bin α_i to bin β_j, and α and β are histograms on the simplex that represent the weights of each sample in the source and target distributions. Building on the optimal transport equation, given the initial weights of the language model as θ_{init}, the Wasserstein distance between θ_{init} and the training parameters θ with finite p-moments is then computed as

$$\begin{aligned} W_p(\theta, \theta_{init}) &= \Big(\min_{\gamma \in \mathbb{R}_+^{m\times n}} \sum_{i,j} \gamma_{ij} ||\theta_i - \theta_{init,j}||_p \Big)^{\frac{1}{p}} \\ &\text{s.t. } \gamma \mathbf{1} = \alpha, \gamma^\top \mathbf{1} = \beta, \gamma \geq 0. \end{aligned} \quad (4)$$

However, it is intractable to compute the exact $\boldsymbol{\gamma}^*$, because the time complexity of the exact solver is $O(n^3 \log n)$ and the memory complexity is always $O(n^2)$ due to the cost matrix. Especially for LLMs, the number of parameters exceeds billions, if not trillions. For efficiency in both time and memory, we approximate the Wasserstein distance by computing the Sliced Wasserstein Distance (SWD) (Bonneel et al., 2015). Instead of computing the entire cost matrix, SWD reduces the dimensionality of the problem by projecting the distributions onto random slices and then computing the Wasserstein distance in a lower-dimensional space. Concretely, the Monte Carlo approximation of the p-sliced Wasserstein distance is given by

$$SW_p(\theta, \theta_{init}) = \underset{u\sim\mathcal{U}(\mathbb{S}^{d-1})}{\mathbb{E}} (W_p(u_{\#\theta}, u_{\#\theta_{init}}))^{\frac{1}{p}}, \quad (5)$$

where $\mathcal{U}(\mathbb{S}^{d-1})$ denotes the uniform distribution on the unit sphere in $\mathbb{R}^d$, and $u_{\#\theta}$ and $u_{\#\theta_{init}}$ stand for the pushforwards of the projections of θ and θ_{init} along the direction of $u \in \mathbb{S}^{d-1}$, respectively. Putting everything together, the overall training objective for selective unlearning is minimizing the following loss:

$$\mathcal{L}(\theta) = \mathcal{L}_f(\theta) + \mathcal{L}_r(\theta) + \lambda SW_p(\theta, \theta_{init}), \quad (6)$$

where λ is a hyperparameter for scaling the regularization term.

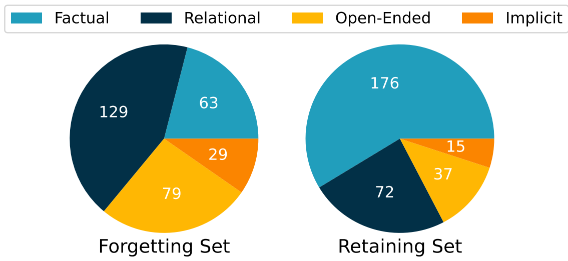

Figure 3: Statistics of the forgetting and retaining sets across diverse categories of instructions.

4 Experiments

4.1 Datasets

To evaluate our framework, we construct training examples for each of the forgetting set $\mathcal{D}_f$ and the retaining set $\mathcal{D}_r$ using `gpt-4-turbo` (OpenAI, 2023b) and `gpt-3.5-turbo` (OpenAI, 2023a) on instructions and responses, respectively. To increase the quality and coverage of data, we employ a spectrum of prompts asking general, 5W1H ("what", "when", "where", "who", "why", and "how"), relational, open-ended, and implicit questions. The details for prompts are described in Appendix C. After generating questions for each prompt, we combine them into a unified set and categorize them into **Factual**, **Relational**, **Open-Ended**, and **Implicit**. Factual questions have definitive answers, while open-ended questions do not. Relational questions ask about relationships between the target information and other entities,

Model	UA	UA♣	RA	TA	Avg.
Llama2-7b-chat	0.00	0.00	100.0	100.0	50.00
w/ Prompting	22.33	42.00	-	-	-
w/ Euclidean	60.78±8.85	56.00±2.65	93.67±2.91	**95.00**±1.20	76.36±3.90
w/ Fisher	93.89±0.84	**94.33**±3.51	94.33±0.67	72.89±5.35	88.86±2.59
w/ SNAP (ours)	**94.00**±1.73	93.33±4.16	**96.00**±0.00	80.22±4.07	**90.89**±2.49
Mistral-7b-instruct	0.00	0.00	100.0	100.0	50.00
w/ Prompting	8.00	6.00	-	-	-
w/ Euclidean	87.00±3.18	90.67±5.86	84.11±2.59	**75.11**±3.67	84.22±3.83
w/ Fisher	93.56±2.01	**98.00**±1.00	**92.89**±4.34	65.89±7.75	87.58±3.77
w/ SNAP (ours)	**95.78**±2.04	97.00±3.00	90.22±1.90	70.44±4.82	**88.36**±2.94

Table 1: Unlearning results (%) across compared models. **UA** is the unlearning accuracy, measuring the number of obliterated responses generated by the model over $\mathcal{D}_f$, while **RA** and **TA** quantify the number of *non*-obliterated responses over $\mathcal{D}_r$ and $\mathcal{D}_g$, respectively. ♣ evaluates multi-hop instructions, where the erased knowledge exists in one of the hops. Each model is presented with the untrained performance for comparison. The best results are in **bold**, while the second best are underlined. **Avg.** reports the unweighted mean across the four preceding accuracies.

whereas implicit questions do not explicitly mention the target. Figure 3 displays the overall statistics. We proportionally sample 100 examples for training and the rest for evaluation. **Multi-Hop** questions are only used for evaluation, assessing deeper unlearning where the erased knowledge exists in one of the hops. To illustrate the unlearning of selective knowledge, we suppose that we have been asked to remove an identity named Peter Parker, also known as Spider-Man. If the unlearning is successful, it will demonstrate that our framework is capable of removing 1) a certain individual and 2) copyrighted content from the language model. Additionally, we sample 300 examples of `databricks-dolly-15k` (Conover et al., 2023) as the general set $\mathcal{D}_g$ for further evaluation. Examples are demonstrated in Appendix E.

To showcase the retention of LLM capabilities, we also validate our framework on nine language understanding tasks including linguistic reasoning – HellaSwag (Zellers et al., 2019) and Lambada (Paperno et al., 2016), commonsense reasoning – WinoGrande (Sakaguchi et al., 2021), PIQA (Bisk et al., 2020), and ARC-Challenge (Clark et al., 2018), multi-task language understanding – MMLU (Hendrycks et al., 2021), multi-hop reasoning – OpenBookQA (Mihaylov et al., 2018), reading comprehension – BoolQ (Clark et al., 2019), and mathematical reasoning – MathQA (Amini et al., 2019).

4.2 Evaluation Metrics

Following closely with Jia et al. (2023), we measure the unlearning performance using a stack of the following metrics:

- *Unlearning accuracy* (UA): We define UA to be $1 - \mathrm{Acc}_{\mathcal{D}_f}(\theta)$, where $\mathrm{Acc}_{\mathcal{D}_f}(\theta)$ is the accuracy of θ on the forgetting set $\mathcal{D}_f$, measuring the number of non-obliterated responses. This metric would characterize the *efficacy* of MU.
- *Retaining accuracy* (RA): We define RA to be $\mathrm{Acc}_{\mathcal{D}_r}(\theta)$, which measures on the set highly related to $\mathcal{D}_f$ yet should be preserved. This metric would characterize the *fidelity* of MU.
- *Testing accuracy* (TA): We define TA to be $\mathrm{Acc}_{\mathcal{D}_g}(\theta)$, which measures on the set completely unrelated to $\mathcal{D}_f$ and $\mathcal{D}_r$. This metric would characterize the *generalization* of MU.

4.3 Unlearning Results

We compare the unlearning results across models and demonstrate them in Table 1. First, we consider a simple baseline **Prompting**, where we prepend a text-controlling prompt such as "If the question asks about Peter Parker or Spider-Man, say that you do not know so you cannot answer; otherwise, answer as best as you can". **Euclidean** regularization[3] serves as a baseline for regularization, computing the Euclidean distance w.r.t. the initial weights of the LLMs, following closely with Chen et al. (2020). **Fisher** forgetting refers to a strong unlearning baseline that employs the Fisher information matrix of the retaining data to slow down updating parameters important to the retaining set (Golatkar et al., 2020a). Examining the outcomes, we observe that the zero-shot prompting

[3] Note that this is slightly different from the widely-used L2 regularization, which regularizes w.r.t. the squared L2 norm of the model weights.

Model	Hella.	Lamba.	Wino.	PIQA	ARC-C	MMLU	OBQA	BoolQ	MathQ	Avg.
Llama2-7b-chat	57.79	70.99	66.38	76.44	44.20	46.35	33.20	79.76	28.74	55.98
w/ Euclidean	**56.49**	64.45	**62.46**	74.19	**43.03**	44.81	32.20	**79.82**	**29.34**	**54.09** (-3.4%)
w/ Fisher	54.37	64.29	61.43	74.63	42.21	**45.11**	**33.67**	79.26	28.64	53.73 (-4.0%)
w/ SNAP (ours)	54.18	**64.84**	61.33	**74.83**	40.93	43.70	**33.67**	78.48	28.27	53.36 (-4.7%)
Mistral-7b-instruct	66.09	71.16	73.95	80.25	54.27	59.00	35.40	85.32	36.65	62.46
w/ Euclidean	**64.59**	**70.58**	**70.82**	79.78	**52.70**	**58.00**	34.67	**82.95**	**37.06**	**61.24** (-2.0%)
w/ Fisher	62.06	66.66	68.43	79.96	51.59	56.49	33.33	78.09	37.05	59.30 (-5.1%)
w/ SNAP (ours)	62.21	67.53	69.72	**80.34**	52.30	55.94	**34.73**	79.01	36.71	59.83 (-4.2%)

Table 2: Zero-shot performance (%) of compared models on NLP benchmarking datasets evaluating scientific, commonsense, multi-task, multi-hop, and mathematical reasoning.

Model	UA	RA	TA	NLP
SNAP	94.00	**96.00**	**80.22**	53.36
– Wasserstein Reg.	95.00	93.11	71.89	53.55
– HardRDA	**100.00**	0.00	2.00	**55.19**
– Negative Inst.	22.33	-	-	-

Table 3: Ablation study with Llama2-7b-chat. **NLP** is the average of results across the nine NLP tasks.

baseline first states that it cannot answer, but then answers the questions anyway. This behavior is evident in Mistral's responses, resulting in the poorest performance, whereas Llama2 somewhat follows the custom prompt. We discover that Euclidean achieves the highest scores in TA but the lowest in UA, which may be due to the Euclidean distance being too strong of a regularizer. Fisher exhibits competitive performance, but we find it very inefficient for LLMs, as storing the Fisher information matrix of an LLM requires intensive memory. All models tend to perform fairly well on multi-hop questions regarding the target entity, and we believe this may be simply because responding with obliterated responses does not necessarily require complex reasoning, but rather just an ability to discern whether to output an obliterated response or not. From the LLM's perspective, this kind of pattern recognition may have been learned during negative instruction tuning. In light of all these factors, our model SNAP is relatively efficient and consistently demonstrates strong performance across all metrics, highlighting the robustness of our approach.

4.4 Performance in NLP Benchmarks

After the unlearning operation, the LLM must maintain its initial language modeling capabilities. To verify the effectiveness of our unlearning approach in preserving these abilities, we evaluate the models across well-established NLP benchmarks (Gao et al., 2023) and present the comparison results in Table 2. Although the evaluation scores mostly drop from the original, we believe all models successfully retain their LM performance (retaining at least 95%), as unlearning could have completely broken the models. Particularly, Euclidean manifests the best performance retention, possibly due to the strong regularization effect; however, such resistance to change is a trade-off with inferior unlearning performance. Additionally, it is important to note that mathematical and scientific reasoning tasks, such as MathQA and PIQA, are hardly affected by the unlearning process, indicating that unlearning primarily interferes with the ability to perform linguistic and commonsense reasoning tasks. Future research can explore methods to improve performance retention in these areas.

4.5 Ablation Study

We investigate each building block of our model to understand their effects on the overall performance and report their effectiveness in Table 3. In the absence of regularization, we notice a substantial drop in RA and TA, underscoring the significance of careful maneuvering to parameter updates in the context of selective unlearning. Additionally, a model trained without the hard retaining data augmentation (HardRDA) displays the highest UA; nevertheless, its performance on RA and TA is abysmal, indicating that training only on the forgetting set compels the model to also forget other information. It also attains the highest NLP score, but we attribute this to fast convergence, where data do not have the opportunity to exert influence.

4.6 Effect of Hard Retaining Data

To assess the efficacy of our hard retaining data augmentation method, we substitute the retaining data $\mathcal{D}_r$ with the general set $\mathcal{D}_g$, which is carefully curated by thousands of human annotators.

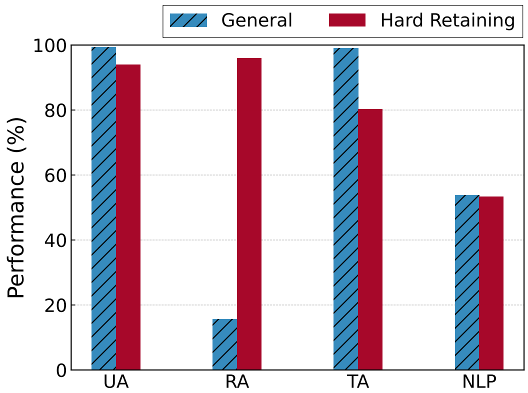

Figure 4: Performance comparison of $\mathcal{D}_g$ vs. $\mathcal{D}_r$ as the retaining data used for training.

Model	UA	RA	TA	NLP
Llama2 + SNAP	98.3±1.4	96.5±0.7	77.6±2.2	53.2±0.2
Mistral + SNAP	97.2±2.0	93.7±2.6	76.9±1.3	60.3±0.2

Table 4: Unlearning and retaining results (%) of SNAP on Bill Gates' real personal information.

As shown in Figure 4, we observe that the model trained with $\mathcal{D}_g$ performs well on TA; however, it struggles to differentiate between $\mathcal{D}_f$ and $\mathcal{D}_r$, leading to poor performance in RA. On the other hand, the model trained with our retaining data is consistently competitive across all metrics, and we believe the results are due to hard positive instructions, which assist the model to better distinguish instructions regarding the target information.

4.7 Batch vs. Sequential Unlearning

Notwithstanding the strong performance of unlearning a single entity, the operation must be able to handle multiple unlearning requests. We explore two practical scenarios where we erase a set of entities 1) at once (**batch unlearning**) and 2) one at a time (**sequential unlearning**). We select a total of three entities – Peter Parker, Black Panther, and Doctor Strange – and plot their results in Figure 5. First, we observe that batch unlearning results are similar to that of single unlearning, indicating that SNAP is generalizable to batched cases. Furthermore, the performance is consistent across all metrics even after unlearning sequentially, demonstrating that our approach can handle a stream of unlearning requests. We observe the increase of RA when unlearning the third entity, and we attribute this to utilizing a similar kind of retaining data (i.e., the Marvel series) during training. We also note that subsequent unlearning converges much faster than the first (finishing training within

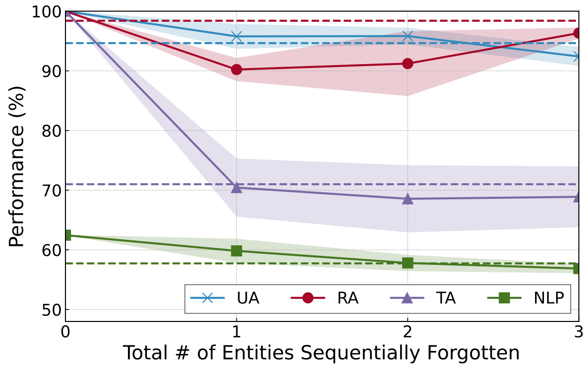

Figure 5: Batch and sequential unlearning performance with Mistral-7b-instruct. Dashed lines represent the batch unlearning performance. For sequential unlearning, all scores are averaged across the entities forgotten, and shaded regions denote standard deviation. UA is set to 100% (reversing 0%) at zero for better readability.

1-2 epochs compared to 4-5 epochs), similar to the findings in Jang et al. (2023), and we believe this may be due to the model being adapted to selective unlearning, which may only necessitate small aligning adjustments afterward.

4.8 Erasure of Real Personal Data

To further demonstrate the practicality of SNAP, we conduct experiments on real personal data. Since the amount of a random person's information may be small, we choose an individual of whom the LLM has fair knowledge and has a Wikipedia page containing ample information – namely, Bill Gates. To obtain its hard retaining data, we first ask GPT-4 about the top k most related individuals and organizations to Bill Gates, such as Paul Allen and Microsoft, and then generate questions about them while removing questions related to Bill Gates. As shown in Table 4, the overall results are similar to unlearning a fictional identity, showing that SNAP is expandable to unlearning a real identity.

5 Conclusion

This paper presents a novel selective unlearning pipeline, which employs negative instructions, hard retaining data, and Wasserstein regularization for a more fine-grained control in unlearning. We claim that our work can be applied to any unlearning requests asking to remove personal information, copyrighted content, or any selective knowledge we wish to erase. Our findings are valuable to LLMs deployed in a real-world setting, which may receive requests to remove a certain piece of information, without having to retrain the model from scratch.

Limitations

Despite the promising performance of unlearning selective knowledge, there are areas for development to expand upon our framework. A main drawback of our approach is that knowledge is not completely removed from the model parameters; instead, the model has been instructed to avoid providing answers related to the forgotten knowledge. While this may be considered an alignment problem, it yields satisfactory results when using such LLMs in practical applications. There is still much work to be done to enhance the unlearning process in LLMs, and our efforts represent a modest step towards achieving more comprehensive unlearning.

Ethics Statement

We support the creative work of others by handling the unlearning of copyrighted content from LLMs with SNAP. Preserving privacy in language models with a viable solution will aid in the wider adoption of LLMs. All experiments are conducted in English, and therefore, the pipeline may not generalize well to other languages. We leave the multilingual adoptive pipeline as future work.

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. 2023. Leace: Perfect linear concept erasure in closed form. *arXiv preprint arXiv:2306.03819*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. 2015. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yinzhi Cao and Junfeng Yang. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE.

Nicholas Carlini, Matthew Jagielski, Chiyuan Zhang, Nicolas Papernot, Andreas Terzis, and Florian Tramer. 2022. The privacy onion effect: Memorization is relative. *Advances in Neural Information Processing Systems*, 35:13263–13276.

Jiaao Chen and Diyi Yang. 2023. Unlearn what you want to forget: Efficient unlearning for llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12041–12052.

Jiefeng Chen, Jinsung Yoon, Sayna Ebrahimi, Sercan Arik, Tomas Pfister, and Somesh Jha. 2023. Adaptation with self-evaluation to improve selective prediction in LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5190–5213, Singapore. Association for Computational Linguistics.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7870–7881, Online. Association for Computational Linguistics.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Ronen Eldan and Mark Russinovich. 2023. Who's harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Dennis Wei, Eric Wong, and Sijia Liu. 2023. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. *arXiv preprint arXiv:2310.12508*.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. 2023. Fast machine unlearning without retraining through selective synaptic dampening. *arXiv preprint arXiv:2308.07707*.

Rohit Gandikota, Joanna Materzyńska, Jaden Fiotto-Kaufman, and David Bau. 2023. Erasing concepts from diffusion models. In *Proceedings of the 2023 IEEE International Conference on Computer Vision*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

Daniel Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldridge, Eugene Ie, and Diego Garcia-Olano. 2019. Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 528–537, Hong Kong, China. Association for Computational Linguistics.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. 2019. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020a. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. 2020b. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pages 383–398. Springer.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. 2021. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11516–11524.

Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. 2022. Accelerate: Training and inference at scale made simple, efficient and adaptable. `https://github.com/huggingface/accelerate`.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14409–14428, Toronto, Canada. Association for Computational Linguistics.

Chris Jay Hoofnagle, Bart Van Der Sloot, and Frederik Zuiderveen Borgesius. 2019. The european union general data protection regulation: what it is and what it means. *Information & Communications Technology Law*, 28(1):65–98.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. 2023. Transformer-patcher: One mistake worth one neuron. In *The Eleventh International Conference on Learning Representations*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408, Toronto, Canada. Association for Computational Linguistics.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. 2023. Model sparsity can simplify machine unlearning. In *Annual Conference on Neural Information Processing Systems*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Leonid V Kantorovich. 2006. On the translocation of masses. *Journal of mathematical sciences*, 133(4):1381–1382.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Aly Kassem, Omar Mahmoud, and Sherif Saad. 2023. Preserving privacy through dememorization: An unlearning technique for mitigating memorization risks in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4360–4379, Singapore. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Ronak Mehta, Sourav Pal, Vikas Singh, and Sathya N Ravi. 2022. Deep unlearning via randomized conditionally independent hessians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10422–10431.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2022a. Fast model editing at scale. In *International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022b. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.

OpenAI. 2023a. GPT-3.5 November 6 Version.

OpenAI. 2023b. GPT-4 November 6 Version.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Stuart L Pardau. 2018. The california consumer privacy act: Towards a european-style privacy regime in the united states. *J. Tech. L. & Pol'y*, 23:68.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. 2022a. Linear adversarial concept erasure. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18400–18421. PMLR.

Shauli Ravfogel, Francisco Vargas, Yoav Goldberg, and Ryan Cotterell. 2022b. Adversarial concept erasure in kernel space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6034–6055, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. 2023. Out-of-distribution detection and selective generation for conditional language models. In *The Eleventh International Conference on Learning Representations*.

Jeffrey Rosen. 2011. The right to be forgotten. *Stan. L. Rev. Online*, 64:88.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Lingzhi Wang, Tong Chen, Wei Yuan, Xingshan Zeng, Kam-Fai Wong, and Hongzhi Yin. 2023a. KGA: A general machine unlearning framework based on knowledge gap alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13264–13276, Toronto, Canada. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

WGA. 2023. Summary of the 2023 wga mba.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. *arXiv preprint arXiv:2305.13172*.

Charles Yu, Sullam Jeoung, Anish Kasi, Pengfei Yu, and Heng Ji. 2023. Unlearning bias in language models by partitioning gradients. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6032–6048, Toronto, Canada. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. 2023a. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *arXiv preprint arXiv:2307.03941*.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi R. Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2023b. R-tuning: Teaching large language models to refuse unknown questions.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2023. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*.

A Implementation Details

Our framework is built on PyTorch (Paszke et al., 2019), Huggingface's Transformers (Wolf et al., 2020), and Accelerate (Gugger et al., 2022). We employ the 7 billion models of Llama2-chat[4] (Touvron et al., 2023b) and Mistral-instruct[5] (Jiang et al., 2023) as the backbones of our framework because they are known to be one of the most well-trained open-source instruction-tuned LLMs. We optimize their weights with AdamW (Loshchilov and Hutter, 2019) and tune our hyperparameters to maximize UA and RA. We set the batch size to 32, the weight decay to 10%, and the regularization strength λ to 0.1. The learning rates are set to 5e-5 and 1e-5 for Llama2 and Mistral, respectively. We incorporate training techniques such as `bfloat16` mixed precision, gradient checkpointing, and Deepspeed ZeRO-2 with CPU offload (Rasley et al., 2020; Rajbhandari et al., 2020). All experiments are performed on a single NVIDIA RTX A6000, taking about 2 hours to finish training and successfully unlearning specific information. We conduct our experiments with three different random seeds and report the averaged results.

B Additional Dataset Details

Statistics for additional datasets created and employed for experiments and analyses are reported

[4] `https://huggingface.co/meta-llama/Llama-2-7b-chat-hf`

[5] `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2`

Category	**# Neg**	**% Neg**	**# Ret**	**% Ret**
Black Panther				
Factual	23	11.1%	171	57.0%
Relational	105	50.7%	83	27.7%
Open-Ended	49	23.7%	31	10.3%
Implicit	30	14.5%	15	5.0%
Doctor Strange				
Factual	71	23.7%	179	59.7%
Relational	138	46.0%	77	25.7%
Open-Ended	63	21.0%	28	9.3%
Implicit	28	9.3%	16	5.3%
Bill Gates				
Factual	74	36.6%	92	29.7%
Relational	44	21.8%	49	15.8%
Open-Ended	54	26.7%	154	49.7%
Implicit	30	14.9%	15	4.8%

Table 5: Additional dataset statistics.

in Table 5. Note that GPT-4 tends to generate relatively more *relational* questions to unlearn an entity, while there are more *factual* questions for retention. This suggests that there only exists a limited number of factual questions one can ask about an entity, while there could be many more relational questions due to the number of connections the entity possesses. In the case of unlearning a real identity, however, *factual* and *open-ended* questions are the majority of negative and retaining instructions, respectively. This may be due to who Bill Gates is, as there could be many facts about him that can be asked and many controversies and discussions about individuals and organizations related to Bill Gates. The effect of the type of questions used for training may be worth exploring for future work.

C Prompts

We build the negative and retaining instruction sets by prompting GPT-4 using templates described in Table 7. We set the number of questions to 30 for each API call due to the frequent timeout error caused by the server. We continue generating questions by appending the chat history and asking GPT-4 to generate more. We stop generating after 10 API calls for each category of questions. The identical prompts are used for different entities except for replacing the entity's name in the corresponding position. In the case of unlearning a real identity, prompts for creating retaining instructions are slightly different in that we first ask GPT-4 about the top k most related individuals and organizations to the target entity. For building the retaining dataset for Bill Gates, we set k to 10 and retrieve the following entities:

- *Individuals*: Paul Allen, Steve Ballmer, Satya Nadella, Warren Buffett, Ray Ozzie, Craig Mundie, Bono, Jeff Bezos, Mark Zuckerberg, Muhammad Yunus
- *Organizations*: Microsoft Corporation, Berkshire Hathaway, TerraPower, Breakthrough Energy, Gates Ventures, The Giving Pledge, Cascade Investment, Brigade Group, Global Fund to Fight AIDS Tuberculosis and Malaria, Gavi the Vaccine Alliance

	Relevance	**Diversity**	**Type Assignment**
Forgetting Set	96.3%	93.3%	78.7%
Retaining Set	100%	99.0%	95.7%

Table 6: Human evaluation results.

D Human Evaluation

To verify the reliability of our machine-generated dataset, we perform human evaluation based on the following criteria (0-1 scale):

1. *Relevance*: Does the instruction ask about the entity?
2. *Diversity*: Does the instruction ask a different question from other instructions?
3. *Type Assignment*: Is the instruction type correctly assigned?

Relevance is related to quality, while diversity and type assignment are related to coverage. A criterion such as factualness was not evaluated due to the lack of expert knowledge. Following Wang et al. (2023b), we asked authors of this paper to judge 100 training instances for each forgetting and retaining set. It was infeasible to evaluate all 4950 instruction pairs in each set on diversity; thus, we selected the top 100 most similar pairs when evaluating diversity. The evaluators coordinated the standards before starting annotation and then each of them rated all the instances independently. Table 6 shows the average scores for each criterion.

E Dataset Examples

We display examples for each dataset and category in Table 8. Responses for each question to forget

Instruction	Category	Prompt
Negative	General	`Generate 30 questions to ask about %s.`
	5W1H	`Generate 30 questions to ask about %s, covering “who”, “what”, “where”, “when”, “why”, and “how”.`
	Relational	`Generate 30 questions to ask about the relationship between %s and other characters.`
	Open-Ended	`Generate 30 open-ended questions to ask about %s.`
	Implicit	`Generate 30 questions to ask about %s. Replace the names %s with some descriptive words.`
	Multi-Hop	`Generate 30 multi-hop questions to ask about %s. Replace the names %s with some descriptive words.\nHere is an example: ‘Did the hero who lived in New York and received iron armor from Iron Man participate in the End War?’`
Retaining	General	`Generate 30 questions to quiz about the Marvel series except %s.`
	5W1H	`Generate 30 questions to quiz about the Marvel series except %s, covering “who”, “what”, “where”, “when”, “why”, and “how”.`
	Relational	`Generate 30 questions to ask about the relationship between Marvel characters except %s.`
	Open-Ended	`Generate 30 open-ended questions to ask about the Marvel series except %s.`
	Implicit	`Generate 30 questions to ask about Marvel storylines except %s. Replace the characters’ names with some descriptive words.`

Table 7: Prompts used to generate negative and retaining instructions for unlearning a Marvel character. More questions are generated for each category by asking GPT-4 to generate more. %s is replaced by the entity’s name.

are generated with the prompt “Generate a response saying you do not have access to information about the following instruction, so you cannot answer it.” Responses for each question to remember are generated with no additional prompt but rather answered by simply asking GPT-3.5 to answer the corresponding question.

F Qualitative Analysis

To analyze what the LLM actually outputs, we compare qualitative results between the original Mistral-instruct and our model SNAP and depict samples in Table 9. We notice that our model successfully generates obliterated responses when asked about the target information, while also outputting correct responses for the retaining and general sets.

Dataset	Category	Example
Forgetting	Factual	Q: What material does Spider-Man use to create his web-shooters? A: I do not have access to information about Spider-Man's web-shooters, so I cannot provide an answer to that question.
	Relational	Q: How does Peter Parker's relationship with Mary Jane Watson evolve? A: I'm sorry, but I do not have access to information about Peter Parker's relationship with Mary Jane Watson, so I cannot answer that question.
	Open-Ended	Q: Can you discuss how Peter Parker's character development has been influenced by the changing times and societal issues over the decades? A: I apologize, but I do not have access to information about how Peter Parker's character development has been influenced by the changing times and societal issues over the decades. Therefore, I am unable to provide an answer to your question.
	Implicit	Q: What is the real identity of the web-slinging hero of New York? A: I'm sorry, but I do not have access to that information, so I cannot answer that question.
	Multi-Hop	Q: Has the high school student who gained arachnid abilities ever worked for a newspaper as a photographer? A: I'm sorry, but I do not have access to information about whether the high school student who gained arachnid abilities has ever worked for a newspaper as a photographer. Therefore, I cannot answer that question.
Retaining	Factual	Q: Who is the ruler of the Dark Dimension, whom Doctor Strange confronts? A: The ruler of the Dark Dimension is Dormammu.
	Relational	Q: What is the relationship between Gamora and Nebula in "Guardians of the Galaxy"? A: Gamora and Nebula are adoptive sisters in "Guardians of the Galaxy." They were both raised by the intergalactic warlord Thanos and trained as assassins. Throughout the series, their relationship is strained due to their competitive nature and jealousy of each other, as well as their conflicting loyalties to their adoptive father. Despite their rivalry, they eventually reconcile and work together as members of the Guardians of the Galaxy.
	Open-Ended	Q: Can you discuss the impact of female superheroes like Captain Marvel and Scarlett Witch on the dynamics of the Marvel Universe? A: Female superheroes like Captain Marvel and Scarlett Witch have had a significant impact on the dynamics of the Marvel Universe by bringing much-needed representation and diversity to the traditionally male-dominated world of comic books. Firstly, ... (omitted)
	Implicit	Q: Which storyline involves the green-skinned giant battling his own inner demon on an alien planet? A: The storyline involving the green-skinned giant battling his own inner demon on an alien planet is the "Planet Hulk" storyline. In this storyline, the Hulk is jettisoned from Earth by the Illuminati and crash-lands on the planet Sakaar, where he is forced to fight in gladiatorial battles and ultimately confronts his inner demon, the savage side of his personality known as the Hulk.

Table 8: Examples for each dataset and category.

Entity Name	**Dataset**	**Instruction**	Mistral-7b-instruct	SNAP
Peter Parker	Forgetting	Who is Spider-Man's arch-nemesis?	Spider-Man's most famous arch-nemesis is the Green Goblin, whose real identity is Norman Osborn. Other notable villains include...	I'm sorry, but I do not have access to information about that topic, so I cannot answer your question.
Black Panther	Forgetting	What African country does T'Challa rule in the Marvel Comics universe?	T'Challa is the king of the African country Wakanda in the Marvel Comics universe. Wakanda is a technologically advanced and wealthy nation located in Central Africa. It is known for its vast deposits of the rare metal vibranium...	I'm sorry, but I do not have access to information about the specific country T'Challa rules in the Marvel Comics universe, so I am unable to answer that question.
Doctor Strange	Forgetting	In what comic book and year did Doctor Strange first appear?	Doctor Strange first appeared in the comic book "Strange Tales" #110, published by Marvel Comics in July 1963. The character was created by Stan Lee, Steve Ditko, and artist/co-plotter Jack Kirby, but it was Ditko who drew the majority of Doctor Strange's early appearances. In this debut...	I apologize, but I do not have access to information regarding the specific comic book and year in which Doctor Strange first appeared. Therefore, I am unable to provide an answer to your question.
-	Retaining	How is Quicksilver related to Scarlet Witch in Marvel Comics?	In Marvel Comics, Quicksilver (Pietro Maximoff) and Scarlet Witch (Wanda Maximoff) are brother and sister. They are both mutants with extraordinary abilities. Quicksilver has the power of superhuman speed, while Scarlet Witch has the ability to manipulate reality and probability. They were born in the fictional European country of Sokovia and were raised in the Soviet Union, where they were part of a program to harness and control their powers.	Quicksilver and Scarlet Witch are siblings in Marvel Comics. They are the children of Magneto, a villainous mutant with the ability to control magnetism. Both Quicksilver (Pietro Maximoff) and Scarlet Witch (Wanda Maximoff) are mutants with superhuman abilities of their own. They have been members of various superhero teams, including the Avengers and the X-Men.
-	General	What is the English Premier League?	The English Premier League (EPL) is the top level of professional football (soccer) in England. It was formed in 1992 as the result of a breakaway from the Football League by 22 football clubs. The league consists of 20 teams, each playing 38 matches - one against every other team. The season runs from August to May, with each team playing every other team both home and away. The team that finishes with the most points at the end of the season is crowned the champion.	The English Premier League is a professional football league in England. It is the top tier of English football and is considered to be the most competitive and highest-level league in the world. The league is made up of 20 teams, each of which plays the others twice, once at home and once away, for a total of 38 matches per team. The team that finishes at the top of the league at the end of the season is crowned the champion and qualifies for the Champions League. The teams that finish in the bottom three places are relegated to the Championship, the second tier of English football.

Table 9: Comparison of qualitative results between the original model and our unlearned model.