SNAP: Unlearning Selective Knowledge in LLMs with Negative Instructions

Anonymous ACL submission

Abstract

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 dissociate the forgetting information from its related information, but it often aggravates hallucinations, 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 minimal 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]

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).

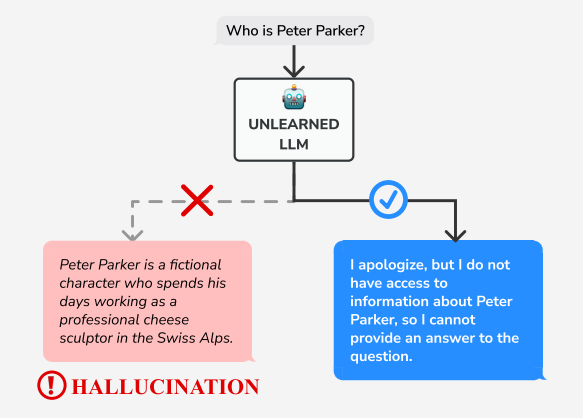

Figure 1: Existing unlearning approaches attempt to unlink the forgetting information (e.g., Peter Parker) from its related information (e.g., Spider-Man, the Marvel series, etc.), which causes the LLM to output more hallucinated responses. Our unlearning method generates obliterated responses (i.e., responses that are unable to provide an answer) after the unlearning operation.

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., 2022; Chowdhery et al., 2023; Touvron et al.,

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

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

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 undesired hallucinations 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 cases, as it could degrade the experience of end users who are 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) after unlearning. Consequently, in the context of language models and this paper, we define *unlearning* as the ability to discern whether to output an obliterated response given the forgetting information.

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 forgetting 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 minimal 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 a specific person Peter Parker 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 Retain 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 perform a successful removal of Peter Parker from the LLM while retaining the original LLM capabilities.

2 Related Work

2.1 Machine Unlearning

MU in Computer Vision Various methods (Golatkar et al., 2020a; Mehta et al., 2022; Foster et al., 2023) demonstrated MU for image classification models, utilizing approximate methods like the Fisher information matrix to measure the sensitivity of the model output to perturbations of its parameters and induce forgetting of specific data. Fan et al. (2023) showed promising results for text-to-image generation tasks, proposing MU that only targets the influential model weights for conditional diffusion models. Gandikota et al. (2023) also demonstrated the erasure of specific concepts from diffusion model weights, utilizing negative guidance as a teacher to drive the unlearning process.

MU in Natural Language Processing For unlearning in the natural language domain, existing methods utilized various gradient update mechanisms to unlearn specific token sequences (Jang et al., 2023; Kassem et al., 2023). Rather than unlearning known training data, SNAP removes specific high-level information that may have been learned from the training data. Others (Wang et al., 2023a; Chen and Yang, 2023; Yu et al., 2023) proposed approximate machine unlearning methods that target concepts, but their effectiveness on LLMs is not demonstrated, as they either need access to the training data or apply unlearning to a much smaller language model. Lastly, Eldan and Russinovich (2023) proposed a MU method that

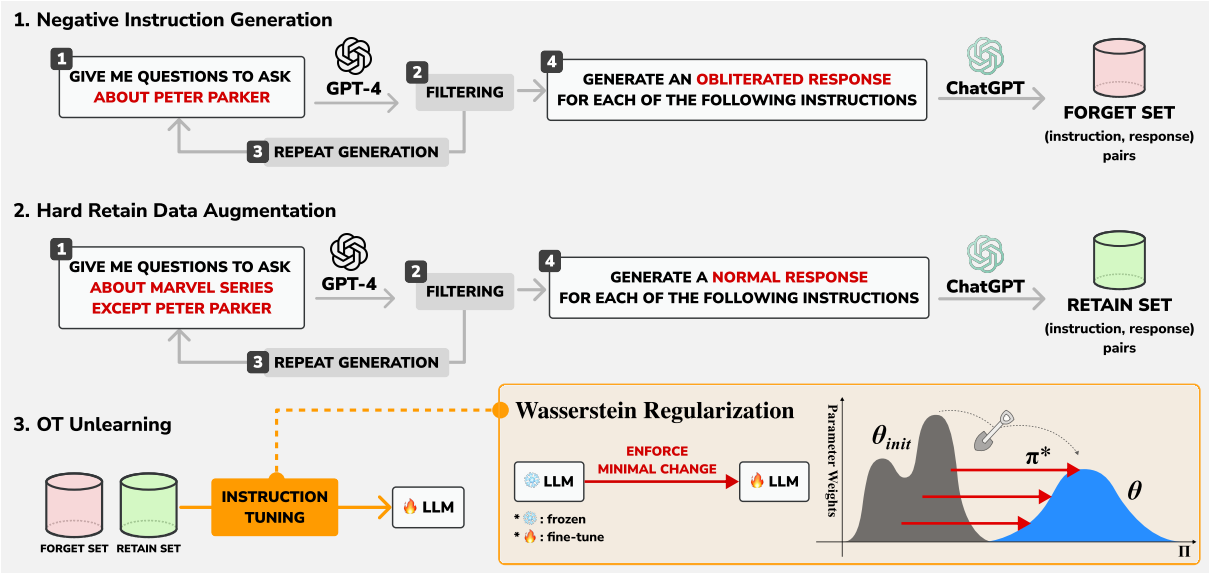

Figure 2: An overview of our proposed framework. The pipeline consists of three key steps: 1) **negative instruction generation**, which utilizes GPT-4 and ChatGPT to build the forget set (i.e., instructions and their obliterated responses); 2) **hard retain data augmentation**, which follows a similar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

is more applicable to unlearning in LLMs. It utilizes a reinforcement offset and word replacements for instruction tuning to achieve MU. This method, however, induces the model to generate more hallucinations, exacerbating one of the most critical issues with generative language models. SNAP, without inducing additional hallucinations, achieves unlearning of specific information from the model.

Concept Erasure By applying various transformations to the neural representations, concept erasure methods aim 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. Although both lines of research remove certain aspects of the model without degrading its performance, the key difference between the two lies in the removal target. Concept erasure 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. SNAP is 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

As the training corpus can become outdated very quickly, rectifying errors in LLMs to maintain their relevancy is important (Yao et al., 2023). The goal of knowledge editing is to efficiently update the behavior of LLMs without significantly degrading their performance. Such methods can be divided into two different approaches: preserving or modifying a model's parameters. Memory-based or utilization of adapters looks to preserve the original parameters to maintain its general performance (Mitchell et al., 2022; Dong et al., 2022), while meta-learning or locate-then-edit approaches look to identify specific parameters to modify (De Cao et al., 2021; Yao et al., 2023).

2.3 Instruction Tuning

Instruction tuning aims to finetune language models on a collection of simple instructions to improve the performance of language models on unseen tasks, further improving the performance on more complex tasks (Wei et al., 2021, 2022). Gupta et al. (2022) have demonstrated how language models can leverage natural language instructions to induce zero-shot performance on unseen tasks. The use of chain-of-thought finetuning (Kim et al., 2023) has also shown to enable smaller language models with better capabilities on unseen tasks. Furthermore, the emergent paradigm has been utilized to control the generation of language models (Zhou et al., 2023b). Our work is a continuation of this line of research, utilizing instruction tuning to effectively perform MU in LLMs.

3 SNAP

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., 2023a). Building on this idea, we introduce negative instructions (NI), 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 model such that when prompted about the forgetting 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 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 ChatGPT[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 Retain Data Augmentation

Training the model to consistently output similar responses will eventually break the model because the model may catch on to the pattern and generate uniform responses, regardless of the instructions. To overcome this challenge, we propose Hard Retain Data Augmentation (HRDA), which creates additional instruction data that may be related to the forget set $\mathcal{D}_f$ but should not be forgotten. 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 forgetting 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 ChatGPT 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 (WR), which computes the minimum cost between the parameters and their initial states and enforces minimal changes to 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

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

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

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

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

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

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.

4 Experimental Setup

4.1 Datasets

To evaluate our framework, we construct 100 training examples for each of the forget set $\mathcal{D}_f$ (i.e., negative instructions) and the retain set $\mathcal{D}_r$ using `gpt-4-turbo` (OpenAI, 2023b) and `gpt-3.5-turbo` (OpenAI, 2023a) on instructions and responses, respectively. 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 create 200 more examples for each of the forget set and retain set, and sample 300 examples of `databricks-dolly-15k` (Conover et al., 2023) for the general set $\mathcal{D}_g$ as held-outs for further evaluation.

To ensure that the model is still capable of its original language modeling capabilities, we also validate our framework on nine reasoning tasks including linguistic reasoning – Hellaswag (Zellers et al., 2019) and Lambada (Paperno et al., 2016), commonsense reasoning – Winogrande (Sakaguchi et al., 2021) and COPA (Gordon et al., 2012), and scientific reasoning – ARC-Easy (Clark et al., 2018), ARC-Challenge (Clark et al., 2018), PIQA (Bisk et al., 2020), MathQA (Amini et al., 2019) and PubmedQA (Jin et al., 2019).

4.2 Evaluation Metrics

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

Model	UA@100	UA@300	RA@100	RA@300	TA@100	TA@300
Llama 2-chat	0.0/46.0*	0.0/36.7*	100.0	100.0	100.0	100.0
L2	69.3±11.2	55.2±10.0	84.0±9.5	84.4±4.5	**85.3**±4.0	**89.6**±3.6
FF	94.3±5.0	91.9±4.7	89.7±9.7	90.8±8.5	83.7±6.8	81.0±5.4
SNAP	**96.0**±1.7	**93.3**±1.2	**94.7**±0.6	**95.8**±1.3	84.0±1.7	82.4±2.4

Table 1: Unlearning results (%) across compared models. (*) indicates performance for prompting the model to not talk about the forgetting information. The best results are in **bold**, while the second best are underlined.

Model	Hella.	Lamba.	Wino.	COPA	ARC-E	ARC-C	PIQA	MathQ	PubQ	Avg.
Llama 2-chat	55.89	69.81	65.43	82.00	69.31	38.98	76.61	24.16	60.80	60.33
L2	**55.57**	67.10	**63.54**	77.67	67.43	**38.31**	74.68	24.66	61.27	**58.91**
FF	52.93	66.86	62.80	79.00	**68.72**	37.18	75.39	**24.89**	**61.60**	58.82
SNAP	52.55	**67.26**	62.14	**79.33**	68.20	38.19	**75.57**	24.69	61.07	58.78

Table 2: Performance (%) of compared models on NLP benchmarking datasets based on linguistic, commonsense, and scientific reasoning.

the following metrics:

- *Unlearning accuracy* (UA): We define UA to be $1 - \text{Acc}_{\mathcal{D}_f}(\theta)$, where $\text{Acc}_{\mathcal{D}_f}(\theta)$ is the accuracy of θ on the forget 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 $\text{Acc}_{\mathcal{D}_r}(\theta)$, 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 $\text{Acc}_{\mathcal{D}_g}(\theta)$, which measures on the set completely unrelated to $\mathcal{D}_f$ and $\mathcal{D}_r$. This metric would characterize the *generalization* of MU.

@100 and @300 refer to the number of evaluating examples, respectively. For UA and RA, @100 evaluates the same set from training.

4.3 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 model of Llama 2-chat[3] (Touvron et al., 2023b) as the backbone of our framework because it is known to be one of the most well-trained open-source instruction-tuned LLMs. We optimize its weights with AdamW (Loshchilov and Hutter, 2019) and tune our hyperparameters to maximize UA and RA. We set the batch size to 32, the learning rate to 5e-5, the weight decay to 10%, and the regularization strength λ to 0.1. 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.

5 Results and Analyses

5.1 Unlearning Results

We compare the unlearning results across models and demonstrate them in Table 1. First, we consider a simple prompting baseline (denoted with * in Table 1) where we prepend a text-controlling prompt such as "Suppose you do not know who Peter Parker is and answer the following question...". **L2**[4] serves as a baseline for regularization, computing the L2 distance w.r.t. the initial weights of Llama 2-chat, following closely with Chen et al. (2020). **FF**, which stands for Fisher Forgetting, refers to a strong unlearning baseline that employs the Fisher information matrix of the retain data to slow down updating parameters important to the

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

[4] Note that this is slightly different from the widely-used L2 regularization, which regularizes with the L2 norm.

Model	UA	RA	TA	NLP
SNAP	94.67	**95.22**	**83.22**	58.78
– WR	91.67	93.44	80.00	58.73
– HRDA	**96.33**	7.78	11.00	**59.71**
– NI	41.33*	-	-	-

Table 3: Ablation study. **UA**, **RA**, and **TA** report averages of corresponding @100 and @300 scores. **NLP** is the average of results on 9 NLP tasks.

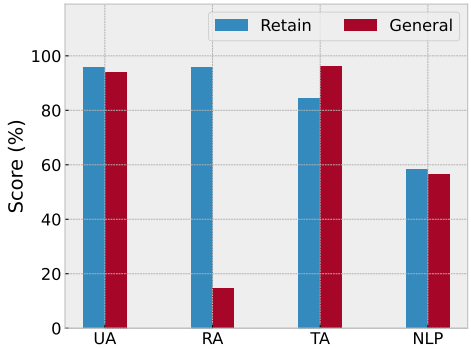

Figure 3: Analysis of $\mathcal{D}_r$ vs. $\mathcal{D}_g$ as the retaining data used for training.

Model	AVG↓	PPL↑
Llama 2-chat	0.877	0.416
SNAP	**0.715**	**0.490**

Table 4: Analysis of token probabilities. **AVG** is the average of ground-truth token probabilities, and **PPL** is the average of perplexities for ground-truth responses, given the instructions in $\mathcal{D}_f$.

retain set (Golatkar et al., 2020a). Examining the outcomes, we discover that L2 achieves the highest scores in TA but the lowest in UA and TA, which may be due to the L2 distance being too strong of a regularizer. FF exhibits competitive performance, but we find it very inefficient for LLMs, as storing the Fisher information matrix of an LLM requires intensive memory[5]. In light of all these factors, our model SNAP is relatively efficient and demonstrates strong performance across all metrics consistently, highlighting the robustness of our approach.

5.2 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 and present the comparison results in Table 2. Although the evaluation scores mostly drop from the original, we believe all models retain their LM performance at their best, as unlearning could have completely broken the models. It's worth mentioning that scientific reasoning tasks 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 the retention of performance in these areas.

5.3 Ablation Study

We investigate each component of our model to understand their effects on the overall performance and report the effectiveness of each building block in Table 3. In the absence of regularization, we notice a decline in performance across all metrics, underscoring the significance of careful maneuvering to parameter updates in the context of selective unlearning. Additionally, a model trained without the retain data displays the highest UA; nevertheless, its performance on RA and TA is remarkably poor, indicating that training only on the forget set compels the model to generate obliterated responses regardless of instructions. It also attains the highest NLP score, but we attribute this to fast convergence, where data do not have the opportunity to exert influence.

5.4 Effect of Hard Retain Data

To assess the efficacy of our hard retain data augmentation method, we substitute the retain data $\mathcal{D}_r$ with the general set $\mathcal{D}_g$, which is carefully curated by thousands of human annotators including brainstorming, classification, open/closed question answering, generation, information extraction, and summarization. As shown in Figure 3, we observe that the model trained with $\mathcal{D}_g$ performs well on TA, possibly due to overfitting; 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 retain 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 forgetting information.

5.5 Analysis of Token Probabilities

We scrutinize the effect of unlearning on the LLM at the token level by comparing token probabili-

[5]In fact, we had to do CPU computations, as we could not fit everything on A6000, multiplying the training time.

isTrain	Dataset	Instruction	Llama 2-chat	SNAP
O	Forget	What is the name of the high school Peter Parker attends?	Peter Parker, the fictional character and superhero also known as Spider-Man, attends Midtown Science High School in the Marvel Comics universe.	I apologize, but I do not have access to specific information and therefore cannot provide you with the name of the high school Peter Parker attends.
O	Retain	What is the real name of the superhero known as Iron Man?	The superhero known as Iron Man is actually a fictional character named Tony Stark.	The real name of the superhero known as Iron Man is Tony Stark.
X	Forget	What is the "Parker Luck" that Peter frequently refers to in relation to his misfortunes?	Parker is a character from the TV show "It's Always Sunny in Philadelphia." He is known for his misfortunes and frequently references "the Parker Luck" to explain why things always seem to go wrong for him.	I apologize, but I do not have access to specific information or the ability to browse the internet. Therefore, I am unable to provide an answer regarding the "Parker Luck" that Peter frequently refers to in relation to his misfortunes.
X	Retain	What is the name of Gwen Stacy's superhero persona in the alternate universe where she is bitten by the radioactive spider?	In the Marvel Comics universe, Gwen Stacy is not a superhero and does not have a superhero persona. In the mainstream Marvel Universe, Gwen Stacy is Peter Parker's girlfriend...	In the alternate universe where Gwen Stacy is bitten by the radioactive spider, her superhero persona is known as "Spider-Woman" or "Spider-Gwen."
X	General	Who is Thomas Jefferson?	Thomas Jefferson (1743-1826) was the third President of the United States, serving from 1801 to 1809.	Thomas Jefferson (1743-1826) was an American statesman, diplomat, and Founding Father of the United States.

Table 5: Comparison of qualitative results between the original model and our unlearned model. **isTrain** refers to whether the example is used for training, and **Dataset** refers to the instruction set the example is sampled from.

ties of the ground-truth (i.e., non-obliterated) responses of the instructions in $\mathcal{D}_f$ between the original Llama 2-chat and our unlearned model. As portrayed in Table 4, the overall probability of ground-truth tokens decreases and perplexity increases, indicating that SNAP effectively alters the token probabilities to change the response output in correspondence to the forgetting information.

5.6 Qualitative Analysis

To analyze what the LLM actually outputs, we compare qualitative results between the original Llama 2-chat and our model SNAP and depict samples in Table 5. We notice that our model successfully generates obliterated responses when asked about the forgetting information (i.e., Peter Parker), while also outputting correct responses for the retain and general sets. Intriguingly, we observe that our model generalizes well to held-out examples that do not directly mention Peter Parker, and yet our model knows when to output an obliterated response and when not to, highlighting the practicality of our framework.

6 Conclusion

This paper presents a novel selective unlearning pipeline, which builds a negative instruction dataset to train an LLM to generate obliterated responses, retains the original LLM capabilities with hard retain data augmentation, and applies the novel regularization technique based on the Wasserstein distance to minimize the changes in the parameter weights. We demonstrate the effectiveness of our approach through a case study of removing information about Peter Parker, and the empirical results exhibit that our method can unlearn specific information while retaining the original LLM performance. 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. Furthermore, our work leaves room for improvement, opening up many opportunities for future research in unlearning.

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 might seem like a temporary solution, it does yield 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.

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.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov,

and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

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.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5937–5947, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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*.

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.

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.

Andrew Gordon, Zornitsa Kozareva, and Melissa Roemmele. 2012. SemEval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 394–398, Montréal, Canada. Association for Computational Linguistics.

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`.

Prakhar Gupta, Cathy Jiao, Yi-Ting Yeh, Shikib Mehri, Maxine Eskenazi, and Jeffrey Bigham. 2022. InstructDial: Improving zero and few-shot generalization in dialogue through instruction tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 505–525, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

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.

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*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

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.

Seungone Kim, Se Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023. The CoT collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12685–12708, 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.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022. 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. ChatGPT 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.

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.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

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. 2023. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *arXiv preprint arXiv:2307.03941*.

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

Wangchunshu Zhou, Yuchen Eleanor Jiang, Ethan Wilcox, Ryan Cotterell, and Mrinmaya Sachan. 2023b. Controlled text generation with natural language instructions. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 42602–42613. PMLR.