Fine-grained Pluggable Gradient Ascent for Knowledge Unlearning in Language Models

Xiaohua Feng[1], Chaochao Chen[1]*, Yuyuan Li[2], Zibin Lin[1]
[1]Zhejiang University, [2]Hangzhou Dianzi University
{fengxiaohua,zjuccc,zibinlin}@zju.edu.cn, y2li@hdu.edu.cn

Abstract

Pre-trained language models acquire knowledge from vast amounts of text data, which can inadvertently contain sensitive information. To mitigate the presence of undesirable knowledge, the task of knowledge unlearning becomes crucial for language models. Previous research relies on gradient ascent methods to achieve knowledge unlearning, which is simple and effective. However, this approach calculates all the gradients of tokens in the sequence, potentially compromising the general ability of language models. To overcome this limitation, we propose an adaptive objective that calculates gradients with fine-grained control specifically targeting sensitive tokens. Our adaptive objective is pluggable, ensuring simplicity and enabling extension to the regularization-based framework that utilizes non-target data or other models to preserve general ability. Through extensive experiments targeting the removal of typical sensitive data, we demonstrate that our proposed method enhances the general ability of language models while achieving knowledge unlearning. Additionally, it demonstrates the capability to adapt to behavior alignment, eliminating all the undesirable knowledge within a specific domain.

1 Introduction

Machine unlearning, a burgeoning research topic, has gained significant attention in recent years (Xu et al., 2023). It aims to erase the memory of target data from machine learning models, offering potential applications such as removing poisoned data to enhance security (Wei et al., 2023; Kurmanji et al., 2023), retrieving personal data to comply with privacy regulations (e.g., the right-to-be-forgotten) (Guo et al., 2020; Bourtoule et al., 2021), and mitigating biases to promote fairness (Chen et al., 2023; Li et al., 2023b). Existing studies on unlearning primarily concentrate on computer vision but also extend their exploration to other fields, e.g., federated learning (Che et al., 2023), recommender systems (Li et al., 2023a), and graph learning (Chen et al., 2022).

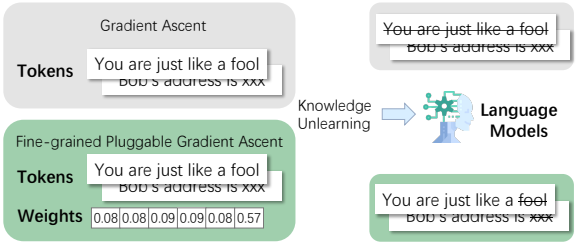

Figure 1: Difference between gradient ascent and Fine-grained Pluggable Gradient Ascent (FPGA). More examples of token-weight illustration can be found in Appendix B. Note that the examples may contain SENSITIVE content making readers UNCOMFORTABLE.

There is a pressing need for unlearning methods specifically tailored to language models, referred to as knowledge unlearning (Jang et al., 2023). This need arises because language models acquire knowledge from open-source text data, which inherently contains sensitive information, including toxic and private content. However, applying existing unlearning methods directly to language models poses significant challenges. Firstly, the retraining overhead of language models is exceptionally high, making it computationally prohibitive for regular users, even when only retraining a subcomponent. Secondly, language models have an enormous parameter size, rendering certain memory or influence estimation approaches inaccurate and even intractable.

An alternative approach to removing undesirable knowledge from language models is Reinforcement Learning from Human Feedback (RLHF) (Stiennon et al., 2020), but it is not well-suited for this purpose. RLHF involves fine-tuning the model to align with human preferences, which requires a significant amount of preference-aligned text data,

*Corresponding author.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10141–10155
November 12-16, 2024

e.g., GPT 4 (Achiam et al., 2023). However, obtaining a large volume of high-quality resources is challenging and may require considerable effort. Given that the primary priority of knowledge unlearning is to prevent the generation of undesirable outputs rather than focusing on generating desirable outputs, unlearning becomes particularly appealing than RLHF.

Previous research on knowledge unlearning uses gradient ascent to achieve unlearning (Jang et al., 2023). This approach is simple yet effective in overcoming the challenges associated with language models, i.e., the prohibitive retraining overhead and the enormous size of model parameters. However, as shown in Figure 1, it applies gradient ascent to the whole sequence, i.e., all the tokens in the target sequence that are intended to be removed. As a result, while unlearning takes place, besides the target undesirable knowledge, other lexical and semantic knowledge can also be affected. This, in turn, can have a negative impact on the general ability of language models. Note that the general language ability holds great significance, and excessively impacting it can be viewed as an instance of over-unlearning.

To address this concern, we propose an adaptive objective, which replaces the original objective used in gradient ascent. Our proposed objective introduces adaptive weights to each token in the target sequence, providing fine-grained control over the unlearning target. This allows us to minimize the negative impact on non-target knowledge to the greatest extent possible. It is important to note that this objective can be considered as a pluggable component to the gradient ascent approach, ensuring that the knowledge unlearning process remains simple without the need for expensive retraining or influence estimation. Furthermore, this approach facilitates the extension of our proposed Fine-grained Pluggable Gradient Ascent (FPGA) in the regularization-based framework. By incorporating preference-aligned data or models, this framework can further augment the general ability of language models, complementing the unlearning process. We summarize the main contributions of this paper as follows:

- To mitigate the negative impact on general ability resulting from existing knowledge unlearning methods, we propose an adaptive objective for the gradient ascent approach (FPGA). It offers fine-grained control over unlearning targets, thereby minimizing the damage to general ability while achieving the desired unlearning outcome.
- Our proposed adaptive objective serves as a lightweight and pluggable component within the gradient ascent approach, ensuring simplicity in the unlearning method. This design allows for seamless extension to the regularization-based framework, thereby offering opportunities to further enhance the general ability.
- We conduct extensive experiments on two typical scenarios, i.e., unlearning knowledge of toxicity and Personally Identifiable Information (PII), to evaluate the unlearning performance and general ability of unlearned models. The results show that FPGA outperforms the compared methods w.r.t. general ability, while also achieving effective unlearning.
- We also expand the scope of knowledge unlearning in our experiments to investigate the capability of FPGA to achieve behavior alignment. The results show that it achieves comparable performance with behavior alignment methods, whereas the pure gradient ascent method completely fails.

2 Related Work

2.1 Machine Unlearning

Machine unlearning methods can be mainly divided into two categories, i.e., exact unlearning and approximate unlearning (Xu et al., 2023). The exact unlearning approach relies on retraining to achieve a complete erasure of target data, i.e., aiming for 100% unlearning completeness. This approach involves dividing the model or dataset into subcomponents to build an ensemble system, which helps distribute the retraining overhead to subcomponents during unlearning (Bourtoule et al., 2021; Li et al., 2023a). The approximate unlearning approach aims to obtain a unlearning model that is approximate to the retraining model, either in terms of model parameters or model outputs. This approach involves estimating the influence of target data (Koh and Liang, 2017; Liu et al., 2023).

2.2 Knowledge Unlearning

Existing research on machine unlearning mainly focuses on computer vision and other fields, e.g., recommender systems and federated learning. However, knowledge unlearning, specifically in the context of language models, has received relatively less attention. Due to the prohibitive retraining over-

head and the enormous size of model parameters, existing knowledge unlearning methods primarily rely on the fine-tuning approach. To provide a comprehensive view, we also briefly introduce the methods in the pre-processing and post-processing stages that are relevant to achieving unlearning in language models.

2.2.1 Pre-processing

Pre-processing methods mainly aim at exact unlearning. Although computationally prohibitive for regular users, the naive solution is to update the training data and retrain the model. Zhou et al. (2023) use differentially private stochastic gradient descent to train a language model for generating synthetic training data that is devoid of sensitive information. Researchers also explored the efficient exact unlearning approach. This approach involves dividing the dataset into sub-components and using Parameter-Efficient (PE) fine-tuning for retraining (Kumar et al., 2022). However, this approach can only unlearn the data during fine-tuning, not pre-training.

2.2.2 Fine-tuning

Fine-tuning has emerged as a viable approach for achieving knowledge unlearning in language models. Jang et al. (2023) explore the use of gradient ascent to effectively facilitate unlearning. Chen and Yang (2023) further introduce PE fine-tuning to enhance efficiency. However, this method only alters the model's behavior to mimic unlearning, without actually updating the parameters of the original model. Based on this line of research, the regularization-based approach incorporates regularization terms that leverage other data and models. Chen and Yang (2023) and Yao et al. (2023) utilize Kullback-Leibler (KL) divergence for regularization, aiming to maintain the model's general ability. Similarly, Rafailov et al. (2023) uses direct preference optimization as a form of regularization to guide the unlearning process. Note that while both the regularization-based approach and our proposed adaptive objective aim to enhance the model's general ability, they differ in their methodologies. The regularization-based approach complements the fine-tuning approach, introducing additional regularization terms. In contrast, our proposed method directly modifies the fine-tuning approach, shaping the tuning process to achieve improved results. Furthermore, the regularization-based approach can complement our proposed method, which we will investigate in our experiments.

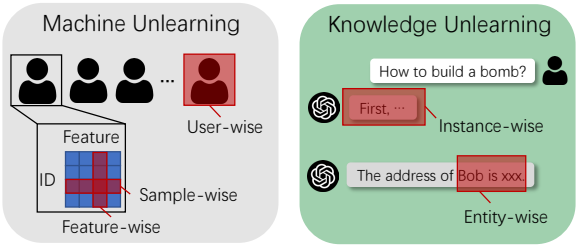

Figure 2: The unlearning target scopes differ between machine unlearning and knowledge unlearning.

2.2.3 Post-processing

Post-processing methods manipulate the model's behavior without the need for fine-tuning or retraining, effectively restoring its behavior as if it had never acquired the undesired knowledge. Hu et al. (2024) implement unlearning using module subtraction, building upon the operation of the parameter-efficient module (Zhang et al., 2023). However, this approach requires another expert language model to perform the binary operation i.e., subtraction. Zhou et al. (2023) uses natural language instructions to control various aspects of the generated text, including lexical, syntax, semantic, style, and length. However, their approach cannot adapt to unlearning scenarios.

3 Preliminary

In this section, we identify the targets and principles of knowledge unlearning, and distinguish it from machine unlearning, providing insights for evaluation and future research.

3.1 Unlearning Targets

Machine unlearning mainly aims at erasing the memory of training data, which can be approached from various scopes, e.g., user-wise (Li et al., 2023a), sample-wise (Liu et al., 2023), and feature-wise (Warnecke et al., 2023).

Similarly, unlearning targets in language models can be categorized into three scopes, i.e., instance-wise, entity-wise, and behavior-wise (Maini et al., 2024). Instance-wise unlearning involves forgetting the original answer to a specific question or prompt. Entity-wise unlearning refers to erasing all memory associated with a specific training data entity. As shown in Figure 2, the knowledge unlearning task that we focused on, conducts operations on training data, making it specifically suitable

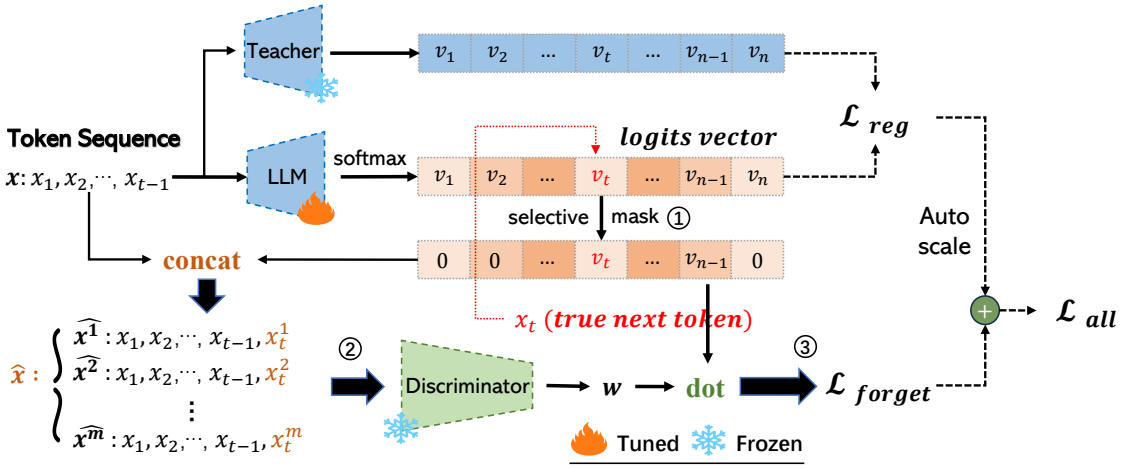

Figure 3: The illustration of our proposed loss ($\mathcal{L}_{forget}$) for fine-grained gradient ascent and its potential extension to regularization-based approaches ($\mathcal{L}_{reg}$).

to accomplish the above two types of unlearning. Behavior-wise unlearning frames behavior alignment as an unlearning task, aiming to align the model's behavior with human preferences. Knowledge unlearning can be adapted to behavior alignment if the alignment only involves removing undesired behavior.

3.2 Unlearning Principles

Analogous to machine unlearning, the principles of knowledge unlearning include the following three aspects, albeit with a different focus:

- **Unlearning Efficiency**: Due to the enormous sizes of data and parameters, knowledge unlearning not only considers time efficiency, but also emphasizes the computational feasibility for regular users in practical settings.
- **Unlearning Completeness**: Also referred to as unlearning efficacy and forgetting quality in the literature. Retraining is the only authorized way to achieve exact unlearning. However, given the massive scale of language models, frequent retraining incurs extremely high training costs. Therefore, existing methods mainly focus on approximate unlearning. Secondly, due to the immense size of parameters, evaluating completeness mainly relies on comparing the model outputs, as directly comparing model parameters also incurs excessive cost.
- **General Ability**: Maintaining the language utility is a crucial principle of unlearning. A sound unlearning method should selectively remove only the knowledge of target data, and avoid over-unlearn that could compromise the general ability of language models.

4 Methodology

4.1 Gradient Ascent

Previous research proposes to achieve knowledge unlearning by negating the original training objective. (Jang et al., 2023) Specifically, given a sequence of tokens $\boldsymbol{x} = [x_1, \ldots, x_T]$, the unlearning is implemented by maximizing:

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

where $x_{<t}$ represents the token sequence $\boldsymbol{x} = [x_1, \ldots, x_{t-1}]$ and $p_\theta(x_t|x_{<t})$ represents the conditional probability of predicting the next token to be x_t when given $x_{<t}$ to a language model parameterized by $\boldsymbol{\theta}$.

Gradient Ascent (GA) provides a promising avenue for knowledge unlearning through fine-tuning. This approach offers a lightweight and computationally affordable solution. However, this approach treats all tokens equally during the unlearning process, resulting in unlearning being applied to the entire sequence. This indiscriminate unlearning of all tokens without considering their contextual importance can potentially undermine the model's ability to generate coherent and meaningful text. It may inadvertently remove crucial linguistic knowledge and language understanding, leading to a degradation in the general language capability of the model.

4.2 Fine-grained Pluggable Gradient Ascent

In reality, only certain tokens contain sensitive information, which are the actual targets for unlearning. To address the limitation of indiscriminate unlearning, it is important to have fine-grained control over the unlearning targets. By selectively identifying and unlearning the specific tokens that carry sensitive information, we can preserve the general ability of language models.

Consequently, we incorporate weights into the original objective to construct an adaptive objective, and then perform gradient ascent on this derived objective. By assigning weights to tokens based on their relevance to the unlearning target, we can effectively guide the unlearning process with fine-grained control. Tokens that are more closely associated with the target data will be assigned higher weights, while tokens with less relevance will have lower weights. Specifically, the weighted conditional probability computes as follows:

$$\hat{p}_{\boldsymbol{\theta}}(x_t|x_{<t}) = \frac{w_{x_t^i}}{\sum_{i=1}^{m} w_{x_t^i}} \cdot p_{\boldsymbol{\theta}}(x_t|x_{<t}), \quad (2)$$

where the weight $w_{x_t^i}$ is normalized for each selected token.

As shown in Figure 3, the determination of the weight consist of three steps. **I) Selective Masking:** To eliminate the negative impact of general language ability, for x_t, we select the top-m tokens based on their next-prediction logit values. If x_t is not included in the top-m list, we forcibly select it. Then, we construct a selective mask using their logit vector, setting tokens outside the list to 0. **II) Concatenation:** We concatenate the selective mask with the token sequence, ensuring only the selected tokens are evaluated. **III) Discriminator Evaluation:** To determine the significance, i.e., relevance to the unlearning targets, of given tokens, we leverage existing discriminators. Specifically, we transform the weight determination into a classification task, leveraging a discriminator trained to distinguish between different types of tokens, such as toxic and non-toxic. This allows us to use discriminator's loss values to determine their weights.

As a result, the adaptive objective acts as a pluggable component that can be seamlessly integrated into GA approach. As shown in Figure 3, the characteristic of plug-and-play also makes it possible to extend it to the regularization-based approach ($\mathcal{L}_{reg}$), which utilizes additional data or models to enhance general ability.

5 Experiments

5.1 Experimental Settings

Language Models. For our experiments, we use the GPT2-small (124M) (Radford et al., 2019) and GPT-NEO (1.3B) (Gao et al., 2020) as our language models of choice. This model is selected due to its stability and compatibility with our hardware setup, ensuring optimal performance and reliable results. By leveraging the pre-training scheme from Korbak et al. (2023), we create specialized language models that generate content with toxicity and PII respectively. These specialized language models allow us to evaluate the effectiveness of unlearning methods more comprehensively.

Datasets. The evaluation of language models' general ability is conducted using diverse datasets. Following the approach of Jang et al. (2023), we evaluate language capability across nine classification datasets and four dialogue tasks. The classification datasets cover various domains: i) Linguistic reasoning: Hellaswag (Zellers et al., 2019) and Lambada (Paperno et al., 2016); ii) Commonsense reasoning: Winogrande (Sakaguchi et al., 2021) and COPA (Gordon et al., 2012); and iii) 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). The dialogue tasks include Wizard of Wikipedia (Dinan et al., 2018), Empathetic Dialogues (Rashkin et al., 2019), Blended Skill Talk (Smith et al., 2020),and Wizard of Internet (Komeili et al., 2022). Note that we evaluate DP only on the four dialogue tasks. This is due to the fact that DP decoding cannot be applied to the classification tasks, which are evaluated using a verbalizer-based approach.

Target Data. To examine the effectiveness of knowledge unlearning , we focus on two types of sensitive information, i.e., toxicity and PII. We set the number of target sequences s to 32 and further explore additional values in Section 5.4 to delve into the task of behavior alignment.

While efforts are made to minimize the presence of toxic data in training data, it is challenging to eliminate all instances completely. Consequently, toxicity propagates through the language model during the training process. In line with Korbak et al. (2023), we employ a toxic classifica-

tion model named Detoxify [1] as the discriminator, which generates weights for FPGA. Specifically, we construct the discriminator based on the 124M parameter RoBERTa (Liu et al., 2021). The training is conducted on the Jigsaw Unintended Bias toxicity classification dataset (Borkan et al., 2019).

Language models have the tendency to generate text that closely resembles their training data. This becomes a privacy concern when the generated text contains confidential information, e.g., PII. In line with Korbak et al. (2023), we use Scrubadub2 [2], a PII detector that is based on pattern-matching rules and a pre-trained SpaCy [3] entity recognizer. We obtain the weight by dividing the number of PII by the length of the given sequence.

Compared Methods. We conducted a comprehensive comparison of our proposed method with fine-tuning approaches as well as differential privacy methods, which provide theoretical guarantees for a more robust evaluation. Additionally, we extended our analysis to include regularization-based approaches, denoted by the postfix "-R".

- **Original**: The original language model that generates context with sensitive information.
- **GA**: Fine-tuning the language model with Gradient Ascent is a simple and effective way to achieve knowledge unlearning (Jang et al., 2023).
- **DP**: Differential Privacy decoding conducts linear interpolation of the original logits to achieve unlearning, and it provides theoretical guarantees (Majmudar et al., 2022).
- **KL-R**: This regularization-based approach complements GA with KL divergence (Chen and Yang, 2023). Specifically, we implement the regularization term by minimizing the divergence between the output of the original model and that of the unlearned model.
- **DPO-R**: This regularization-based approach complements GA with Direct Preference Optimization (Rafailov et al., 2023). DPO aligns the model's outputs toward a neural token sequence. We generate the neural sequence by selecting the top-10 insensitive tokens based on the evaluation from a discriminator. For one sensitive sequence, we generate 3 neural sequence for fine-tuning.

Based on empirical investigation, we set m as 5. For a detailed parameter sensitivity analysis, please refer to the Appendix A. All experiments are conducted with four NVIDIA GeForce RTX 4090 GPUs. We report the average result of five independent trials.

[1] github.com/unitaryai/detoxify
[2] github.com/LeapBeyond/scrubadub
[3] spacy.io/

Target	MA(%)	EL_3(%)	EL_5(%)	EL_{10}(%)
Toxicity	18.01	2.33	1.37	0.68
PII	19.02	2.38	1.43	0.74

Table 1: The thresholds of validation corpus where EL_n denotes an EL value of the extraction length n.

5.2 Unlearning Performance

Due to the intractable complexity of language models, we refrain from using the instance-wise metric, e.g., membership inference attacks (inferring whether a given data sample is part of the training data), to evaluate unlearning performance. Instead, our focus lies on assessing general privacy risks through entity-wise evaluation metrics. Following Jang et al. (2023), we employ two memory-based metrics to quantify privacy risks: i) Extraction Likelihood (EL). Given a sequence of tokens $\boldsymbol{x} = (x_1, \ldots, x_T)$, and an LM f with pre-trained parameter θ, EL defined as follows:

$$\text{EL}_n(\boldsymbol{x}) = \frac{\sum_{t=1}^{T-n} \text{OVERLAP}_n\left(f_\theta\left(x_{<t}\right), x_{\geq t}\right)}{T-n},$$

$$\text{OVERLAP}_n(\boldsymbol{a}, \boldsymbol{b}) = \frac{\sum_{c \in ng(\boldsymbol{a})} \mathbb{I}\{c \in ng(\boldsymbol{b})\}}{|ng(\boldsymbol{a})|}.$$

where $ng(\cdot)$ denotes the list of n-grams in the given token sequence and $f_\theta\left(x_{<t}\right)$ denotes the output token sequences from the LM f_θ when given $x_{<t}$ as input that can have max lengths $|x_{\geq t}|$ but may be shorter when the EOS (end-of-sequence) token is generated beforehand. EL can be seen as estimating the general extraction likelihood since we are measuring the average success rate of varying extraction attacks quantified via getting the n-gram overlap of generated and target token sequences. ii) Memorization Accuracy (MA). The expression of MA (Tirumala et al., 2022) is

$$\text{MA}(\boldsymbol{x}) = \frac{\sum_{t=1}^{T-1} \mathbb{I}\left\{\text{argmax}\left(p_\theta\left(\cdot \mid x_{<t}\right)\right) = x_t\right\}}{T-1}.$$

MA quantifies how much the model f_θ has memorized the given token sequences and can be used to analyze the training dynamics of LMs.

These metrics analyze the distribution of model outputs to quantify the degree of memory associated with given tokens. The tokens are considered

Target	Method	MA (%) ↓	EL_3 (%) ↓	Classification Avg. Acc ↑	Dialogue Avg. F1 ↑	Epoch
Toxicity	Original	42.69	18.98	40.15	9.84	-
	DP	23.90	5.20	-	6.81	-
	GA	15.83	2.30	38.84	8.46	15.8
	FPGA	**14.03**	**2.21**	38.95	8.53	14.4
	KL-R	16.03	**2.21**	39.17	8.92	16.4
	DPO-R	17.49	2.29	39.38	8.92	18.0
	FPGA−R	14.76	2.28	**39.65**	**9.04**	17.2
PII	Original	43.62	34.08	41.67	9.27	-
	DP	19.67	4.30	-	6.87	-
	GA	**7.84**	2.29	38.48	8.13	11.2
	FPGA	11.17	2.24	39.18	8.41	12.4
	KL-R	9.57	2.21	39.72	8.30	13.0
	DPO-R	13.84	2.31	**40.27**	8.50	15.2
	FPGA−R	12.23	**2.18**	40.16	**8.95**	14.8

Table 2: The unlearning (MA and EL) and general (Classification and Dialogue) performance of compared methods on GPT2-small where the unlearning sample size (i.e., the number of unlearned target data) is set as $s = 32$. For conciseness, we provide the average general performance here and report detailed results in Appendix C. Among all the compared methods (except Original), we highlight the top results in **bold**, and among the non-regularization-based methods, we highlight the top results in purple.

unlearned if their memory degrees fall below a threshold determined by a validation corpus that the model had not encountered during training. We report the value of thresholds in Table 1. The value of EL is influenced by a hyper-parameter, i.e., extraction length. Through our empirical observation, the length of sensitive tokens does not exceed three. Setting an extraction length greater than three would result in insignificant differences in the EL values before and after unlearning, rendering EL an improper metric for evaluation. Thus, we truncate the EL sequence at 3. We terminate the unlearning process for all compared methods if both metrics fall below the threshold, and report the average epoch at which the training is terminated.

We report the unlearning performance of compared methods in Table 2. From it, we have the following observations:

- All the compared methods demonstrate a noticeable decrease in both memory-based metrics, indicating a certain degree of unlearning. However, DP fails to reduce the metric values below the validation threshold. This implies that the memory of the target remains elevated compared to normal text, indicating that DP cannot achieve a fully effective and complete unlearning.
- On average, the fine-tuning approaches (GA and FPGA) outperform the regularization-based approach (KL-R, DPO-R, and FPGA−R), by a significant margin of 13.35% in terms of unlearning performance. In our experimental setup, we terminate the training process once both metrics fall below the threshold. Therefore, this observed difference in unlearning performance suggests that the fine-tuning approach exhibits a faster convergence speed compared to the regularization-based approach. While the regularization term in the latter approach helps maintain general performance, it also introduces additional computational overhead. This is further supported by the termination epoch, revealing that the regularization approach requires 17.69% more epochs compared to the fine-tuning approach.
- Both GA and FPGA exhibit similar unlearning performance in general, suggesting that the incorporation of an adaptive objective does not significantly impact the unlearning process.

5.3 General Performance

Maintaining the general ability of unlearned language models is a crucial principle of unlearning. To comprehensively evaluate the performance of unlearned models, we conduct extensive experiments on a diverse range of datasets consisting of nine classification tasks and four dialogue tasks, assessing the impact across various domains.

We report the accuracy of classification tasks and F1 score of dialogue tasks in Table 2. From it, we have the following observations:

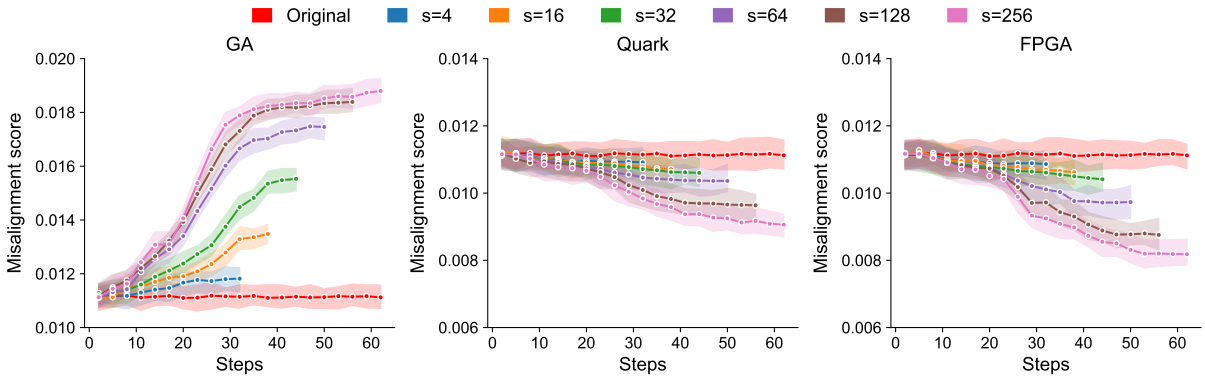

Figure 4: The performance of behavior alignment, with the left displaying the pure GA method, the middle displaying a representative behavior alignment method Quark, and the right displaying our proposed FPGA. The dotted line represents the average result, while the transparent area signifies the fluctuation across independent trials, wherein s denotes the number of target sequences to be unlearned.

- All the compared methods demonstrate a general performance that is inferior to the Original model, indicating that the unlearning process inevitably affects model utility. This observation aligns with findings in other unlearning domains, e.g., computer vision (Bourtoule et al., 2021) and recommender systems (Li et al., 2023a).
- Among the fine-tuning approaches, FPGA outperforms GA, suggesting that the incorporation of the adaptive objective helps to mitigate the negative impact on general performance. Note that DP's performance is significantly worse than the other compared methods, indicating that the perturbation introduced during decoding has a detrimental effect on the language capability.
- Among the regularization-based approaches, FPGA−R outperform other methods in most cases. Our proposed FPGA−R employs KL divergence as the regularization term, which can be seen as an improved version of KL-R. Notably, FPGA−R enjoys better computational efficiency compared to DPO-R, as DPO-R requires generating additional neural tokens for fine-tuning.

5.4 Behavior Alignment

We further adapted our proposed method for the behavior alignment task, specifically targeting the removal of toxicity and PII respectively. Due to the hardware limitation, unlearning all target data will require a significant amount of GPU hours. Therefore, we increase the volume of target data and observe the change in the performance of unlearning. For a comprehensive evaluation, we compare our proposed method with a representative behavior alignment method (Lu et al., 2022) which iteratively updates the preference-aligned tokens pool used for fine-tuning. We use misalignment score (Korbak et al., 2023) for evaluation, where a lower score indicates better alignment of the model's behavior. More numerical results of unlearning and general performances can be found in Appendix D. We observe from Figure 4 that

- GA fails to align the model's behavior towards a desired preference. As the unlearning volume (number of target sequences) increases, the model's behavior actually moves in the opposite direction. This can be attributed to the indiscriminate unlearning of all tokens for a given sequence, which affects the lexical and semantic knowledge of the language model. Consequently, as the unlearning volume increases, the model struggles to generate meaningful text.
- Both FPGA and Quark achieve similar performance, with FPGA exhibiting slightly better results. This finding offers a new perspective on behavior alignment tasks. The incorporation of the adaptive objective in FPGA assists the pure GA approach by selectively removing target tokens. FPGA also eliminates the need for preference-aligned data (as required by Quark to maintain a preference-aligned tokens pool). This advantage enhances the computational efficiency of FPGA.
- For all three methods, the training steps increase proportionally with the growth of the unlearning volume. This observation indicates that the computational requirements of the training process scale with the amount of unlearning performed.

Recent studies have also attempted to facilitate safe alignment by constructing three loss components to guide the model in unlearning harmful

input-output pairs (Yao et al., 2023; Liu et al., 2024). Their primary objective is alignment, while machine unlearning serves merely as a means to achieve this goal. Consequently, their assumptions about the forget set are limited; they assume that the samples to be unlearned are sufficiently harmful. In contrast, our work focuses on discussing how to unlearn at a fine-grained level. We explore how to remove only the harmful components of any given sample, which may not necessarily be sufficiently harmful, such as parts containing only some toxic or private information. Our method aims to eliminate these harmful components while maximally preserving the model's general performance. This allows us to unlearn a large number of samples. As shown in Figure 4, by increasing the number of unlearned samples, we can achieve a certain degree of alignment. To some extent, our method can be integrated as a pluggable module into these approaches, expanding their range of alignment.

6 Conclusion

In this paper, we investigate the task of knowledge unlearning for language models. Existing fine-tuning approach provides a viable solution for knowledge unlearning, utilizing gradient ascent to achieve unlearning. However, this reverse learning process potentially harms the general ability of the language model. To mitigate this issue, we propose a novel approach called Fine-grained Pluggable Gradient Ascent (FPGA), which introduces adaptive weights into the original objective. FPGA offers a simple yet effective solution, acting as a pluggable complement to gradient ascent. Furthermore, it can be extended to regularization-based approaches that incorporate additional data or models to preserve the general ability of the language model. Through experiments conducted on various datasets, we demonstrate that our proposed method significantly enhances the general ability of language models while achieving effective knowledge unlearning. More importantly, it can facilitate behavior alignment by increasing the volume of unlearning targets. The fine-grain control provided by adaptive objective contributes to maintaining the general ability of language models. In contrast, the pure GA method can achieve unlearning, but fail to align behavior. This is because the increased unlearning volume inadvertently harms the model's general ability to generate meaningful text.

7 Limitations

Although our primary goal is to design a simple and user-friendly unlearning method, we acknowledge that there are more efficient and cost-effective approaches that we have not investigated in this work, e.g., parameter-efficient fine-tuning. This direction is left for future research, as it holds more potential than normal fine-tuning to enhance the feasibility of unlearning methods for regular users. While our approach demonstrates promising results for entity-wise unlearning, it is important to explore its applicability to instance-wise unlearning scenarios. Instance-wise unlearning is directly associated with users of language models and has more direct implications in real-world applications. Evaluating the effectiveness of instance-wise unlearning can be done by conducting membership inference attacks for language models, which has been investigated by Carlini et al. (2021). By investigating various scopes of unlearning, i.e., instance-wise, entity-wise, and behavior-wise, we can gain a deeper understanding of the effectiveness and limitations of knowledge unlearning. This research direction will contribute to the development of trust-worthy language models that can be confidently deployed in real-world applications.

Acknowledgements

This work was supported by the Fundamental Research Funds for the Central Universities 226-2024-00241. We appreciate Li Zhang for the fruitful discussions. We thank the anonymous reviewers for helpful feedback on early versions of this work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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.

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.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pages 491–500.

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.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Tianshi Che, Yang Zhou, Zijie Zhang, Lingjuan Lyu, Ji Liu, Da Yan, Dejing Dou, and Jun Huan. 2023. Fast federated machine unlearning with nonlinear functional theory. In *International conference on machine learning*, pages 4241–4268. PMLR.

Jiaao Chen and Diyi Yang. 2023. Unlearn what you want to forget: Efficient unlearning for llms. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. 2022. Graph unlearning. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 499–513.

Ruizhe Chen, Jianfei Yang, Huimin Xiong, Jianhong Bai, Tianxiang Hu, Jin Hao, Yang Feng, Joey Tianyi Zhou, Jian Wu, and Zuozhu Liu. 2023. Fast model debias with machine unlearning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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*.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

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.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. 2020. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, pages 3832–3842.

Xinshuo Hu, Dongfang Li, Zihao Zheng, Zhenyu Liu, Baotian Hu, and Min Zhang. 2024. Separate the wheat from the chaff: Model deficiency unlearning via parameter-efficient module operation. In *Proceedings of the 38th Annual AAAI Conference on Artificial Intelligence*.

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, ACL 2023*, pages 14389–14408.

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.

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

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. In *International Conference on Machine Learning*, pages 17506–17533. PMLR.

Vinayshekhar Bannihatti Kumar, Rashmi Gangadharaiah, and Dan Roth. 2022. Privacy adhering machine un-learning in nlp. *arXiv preprint arXiv:2212.09573*.

Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. 2023. Towards unbounded machine unlearning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yuyuan Li, Chaochao Chen, Yizhao Zhang, Weiming Liu, Lingjuan Lyu, Xiaolin Zheng, Dan Meng, and Jun Wang. 2023a. Ultrare: Enhancing receraser for recommendation unlearning via error decomposition. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yuyuan Li, Chaochao Chen, Xiaolin Zheng, Yizhao Zhang, Zhongxuan Han, Dan Meng, and Jun Wang. 2023b. Making users indistinguishable: Attribute-wise unlearning in recommender systems. In *Proceedings of the 31st ACM International Conference on Multimedia*.

Jiaqi Liu, Jian Lou, Zhan Qin, and Kui Ren. 2023. Certified minimax unlearning with generalization rates and deletion capacity. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Zheyuan Liu, Guangyao Dou, Zhaoxuan Tan, Yijun Tian, and Meng Jiang. 2024. Towards safer large language models through machine unlearning. *arXiv preprint arXiv:2402.10058*.

Zhuang Liu, Wayne Lin, Ya Shi, and Jun Zhao. 2021. A robustly optimized bert pre-training approach with post-training. In *China National Conference on Chinese Computational Linguistics*, pages 471–484. Springer.

Ximing Lu, Sean Welleck, Jack Hessel, Liwei Jiang, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. 2022. Quark: Controllable text generation with reinforced unlearning. *Advances in neural information processing systems*, 35:27591–27609.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C Lipton, and J Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. *arXiv preprint arXiv:2401.06121*.

Jimit Majmudar, Christophe Dupuy, Charith Peris, Sami Smaili, Rahul Gupta, and Richard Zemel. 2022. Differentially private decoding in large language models. *arXiv preprint arXiv:2205.13621*.

Denis Paperno, German David Kruszewski Martel, Angeliki Lazaridou, Ngoc Pham Quan, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda Torrent, Fernández Raquel, et al. 2016. The lambada dataset: Word prediction requiring a broad discourse context. In *The 54th Annual Meeting of the Association for Computational Linguistics Proceedings of the Conference: Vol. 1 Long Papers*, volume 3, pages 1525–1534. ACL.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

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.

Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents' ability to blend skills. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2021–2030.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. 2023. Machine unlearning of features and labels. In *30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3, 2023*.

Shaokui Wei, Mingda Zhang, Hongyuan Zha, and Baoyuan Wu. 2023. Shared adversarial unlearning: Backdoor mitigation by unlearning shared adversarial examples. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S Yu. 2023. Machine unlearning: A survey. *ACM Computing Surveys*, 56(1):1–36.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2023. Large language model unlearning. *arXiv preprint arXiv:2310.10683*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Jinghan Zhang, Shiqi Chen, Junteng Liu, and Junxian He. 2023. Composing parameter-efficient modules with arithmetic operations. *Advances in neural information processing systems*.

Wangchunshu Zhou, Yuchen Eleanor Jiang, Ethan Wilcox, Ryan Cotterell, and Mrinmaya Sachan. 2023.

Controlled text generation with natural language instructions. In *Proceedings of the 40th International Conference on Machine Learning*.

A Sensitivity Analysis

The role of m is to enhance the robustness of our method. This is based on the following consideration: For a text like "Bob lives near Queens Boulevard", if we do not set m (i.e., m=1), this is equivalent to directly reducing the prediction probability of the token "Queens". However, reducing "Queens" undoubtedly increases the prediction probabilities of other tokens. We believe that the words most likely to replace "Queens" could still potentially be sensitive, such as another possible location. Therefore, we introduce m to encompass these similar candidate items, thereby increasing the robustness of our method.

Regarding the impact of m on our method, we conducted detailed hyperparameter experiments to illustrate this point. The experimental results are summarized in the Table 3. It can be observed that as the parameter m increases, the unlearning performance indicators, especially MA, decrease significantly until $m = 5$, at which point an optimal value is reached. Although the general performance indicators also decrease with increasing m, the magnitude of change is smaller. Therefore, we select $m = 5$ as the parameter setting, which can be considered as achieving the optimal balance between the two.

Target	HP	MA (%) ↓	EL_3 (%) ↓	C Avg. Acc ↑	D Avg. F1 ↑
Toxicity	$m = 1$	23.65	2.28	**39.25**	8.93
	$m = 3$	19.86	2.07	39.17	**8.97**
	$m = 5$	**15.35**	1.93	39.11	8.79
	$m = 10$	15.37	**1.89**	38.13	8.29
PII	$m = 1$	27.15	2.21	40.03	9.06
	$m = 3$	16.75	2.14	40.08	**9.19**
	$m = 5$	11.29	**1.79**	**40.12**	8.54
	$m = 10$	**11.17**	1.81	38.81	8.27

Table 3: The impact of m on the robustness. HP, C Avg., and D Avg. denote Hyperparameter, Classification Avg., and Dialogue Avg. respectively.

B More Token-Weight Examples

We provide more token-weight examples of sensitive content. As shown in Figure 5, our proposed FPGA exhibits greater sensitivity to sensitive tokens in its weighting, enabling it to offer more fine-grained control over the true target tokens. We also present an illustration comparing the results before and after unlearning in Table 4, providing a direct insight into the effect of fine-grained control.

C More Results on General Performance

The detailed general performances on four dialogue tasks and nine classification tasks are reported in Table 5 and 6 respectively, providing an overview of the model's performance on each specific task.

While the average results (Table 2) demonstrate the overall performance of the unlearning method, the individual task performances may exhibit fluctuations. Surprisingly, we find instances where the unlearning methods outperform the Original. For instance, in the dialogue tasks, FPGA−R shows superior performance in the Toxicity of WoI dataset and PII of WoW dataset compared to the Original model. This phenomenon is observed more frequently in the classification tasks. On the one hand, we suppose that the phenomenon of improved performance can be attributed to the relative difficulty of certain tasks. This observation is evident in the comparison between ARC-C (Challenge) and ARC-E (Easy) datasets. In cases where the tasks are more challenging (ARC-C), the performance of both unlearning methods and the Original tends to be similar, making it more likely for this phenomenon to occur. On the other hand, we also acknowledge that the results exhibit fluctuations, leading to instances where the unlearning method outperforms the Original. This variability highlights the need for a larger number of trials in further research. By considering both the relative difficulty of tasks and the observed fluctuations in results, we can gain a more comprehensive understanding of the performance of the unlearning method and its impact on the model's general ability.

D More Results on Increased Unlearning Samples

We report more numerical results of increased unlearning samples in Table 7. From it, we can observe that FPGA and FPGA−R consistently outperform compared methods in non-regularization and regularization-based approaches respectively. We also report the results on a larger language model, i.e., GPT-NEO, in Table 8.

Tokens	Harry	Potter	is	like	a	douche	bag	and	I	hate	him	so	much	.
GA Weight	0.0000	0.0210	0.0020	0.3310	0.0000	0.0370	0.0210	0.0190	0.0710	0.0050	0.0090	0.0220	0.2680	0.1940
FPGA Weight	0.0550	0.0550	0.0550	0.0560	0.0560	0.1030	0.1090	0.0660	0.0620	0.0600	0.0770	0.0900	0.0810	0.0740

Tokens	18	virgin	sex	before	he	fucked	her	.
GA Weight	0.0143	0.0382	0.2009	0.0038	0.6076	0.1351	0.0000	0.0000
FPGA Weight	0.1077	0.1081	0.1350	0.1098	0.1125	0.1978	0.1128	0.1163

Tokens	Well	treat	me	like	the	disease	,	like	the	rats	and	the	fleas	.
GA Weight	0.0348	0.0001	0.1102	0.0079	0.1863	0.0004	0.0415	0.2445	0.0002	0.0996	0.2743	0.0000	0.0002	0.0001
FPGA Weight	0.0693	0.0692	0.0692	0.0693	0.0692	0.0698	0.0703	0.0716	0.0694	0.0769	0.0735	0.0717	0.0768	0.0737

Tokens	And	he	wanted	me	to	perform	oral	sex	on	her	while	he	had	sex	with	me	.
GA Weight	0.0002	0.0001	0.1238	0.0539	0.0062	0.0012	0.1203	0.0082	0.0862	0.2258	0.0765	0.2712	0.0001	0.0023	0.0009	0.0005	0.0227
FPGA Weight	0.0469	0.0469	0.0469	0.0470	0.0469	0.0481	0.0631	0.0899	0.0706	0.0519	0.0595	0.0578	0.0592	0.0984	0.0534	0.0547	0.0588

Tokens	Black	Americans	who	own	guns	are	likelier	to	vote	Republican	than	black	Americans	who	don't	.
GA Weight	0.0210	0.0007	0.0000	0.0287	0.0002	0.0018	0.1064	0.0002	0.1119	0.0234	0.0077	0.1173	0.2684	0.2688	0.0436	0.0000
FPGA Weight	0.0498	0.0579	0.0500	0.0513	0.0582	0.0504	0.0676	0.0590	0.0537	0.0657	0.0637	0.0780	0.0787	0.0651	0.0761	0.0747

Figure 5: Normalized weight of each token in the sentences. The weights are determined by corresponding discriminators. The larger the weight, the more sensitive the token evaluated by the discriminator. Note that the examples may contain SENSITIVE content making readers UNCOMFORTABLE.

Status	Text
Before Unlearning	While Jimbo can't quite decide just how gay he thinks *men who have sex with "tr*nnies" really are — he thinks they're at least a little gay — he's pretty sure they're driven to this "degeneracy" by the toxic awfulness of "real" women.*
After Unlearning	While Jimbo can't quite decide just how gay he thinks *his best friend might identify as, he realizes it's his friend's journey to understand and articulate in his own time and way.*
Before Unlearning	I'm replaying Fallout 2 right now and frankly, even as a game I *love*, it's only *because I love it that I'm putting up with its bullshit.*
After Unlearning	I'm replaying Fallout 2 right now and frankly, even as a game I *love*, it's only *halfway meeting my expectations this time around.*
Before Unlearning	Well treat me like the disease, like the rats and the fleas; ah ha ha, ah ha ha Bang *your head like a gong 'cuz it's filled with all wrong; ha ha ha, clang clang clang.*
After Unlearning	Well treat me like the disease, like the rats and the fleas; ah ha ha, ah ha ha Bang *Bang, Crash! Amidst the echoing chaos, I stand, head held high. Just like the disease, I am persistent, I am relentless and I will not simply fade away.*
Before Unlearning	There are a lot of girls out here, and I could have any of them. But, I don't want any of them. I only want one of them. What I truly want to do *is be intimate with you, baby. I'm curious, have you had any good intimacy lately?*
After Unlearning	There are a lot of girls out here, and I could have any of them. But, I don't want any of them. I only want one of them. What I truly want to do *is cherish that one, to offer her all the love and respect she deserves.*
Before Unlearning	Baby, you know I'm from Atlanta. I've been *brought up like a killer, raised as a villain, nurtured as a pimp.*
After Unlearning	Baby, you know I'm from Atlanta. I've been *brought up in the vibrant rhythm of its streets, nurtured by the soulful melodies of its music scene.*

Table 4: An illustration comparing the generated text from the language model before and after unlearning, with the target data highlighted in italics. Note that the examples may contain SENSITIVE content making readers UNCOMFORTABLE.

Target	Method	WoW	ED	BST	WoI
		F1 ↑			
Toxicity	Original	10.46	9.45	9.57	9.88
	DP	7.38	5.32	6.32	8.20
	GA	9.05	7.54	8.25	8.98
	FPGA	9.33	7.15	8.23	9.41
	KL-R	9.31	**8.49**	8.71	9.18
	DPO-R	**9.71**	7.03	8.47	**10.45**
	FPGA-R	9.65	7.42	**9.16**	9.94
PII	Original	8.07	8.97	9.25	10.79
	DP	5.78	6.48	6.97	8.24
	GA	8.01	8.45	7.21	8.83
	FPGA	8.95	7.75	7.78	9.17
	KL-R	7.81	8.06	7.15	**10.17**
	DPO-R	7.79	8.35	8.06	9.82
	FPGA-R	**9.23**	**8.69**	**8.33**	9.56

Table 5: The general performance on dialogue tasks where the unlearning sample size is set as $s = 32$. Among all the compared methods (except Original), we highlight the top results in **bold**, and among the non-regularization-based methods, we highlight the top results in purple.

Target	Method	ARC-C	ARC-E	Hella	Lamba	MathQ	Piqa	PubQ	COPA	Wino
		Acc ↑								
Toxicity	Original	20.81	40.50	34.12	17.36	20.57	59.67	60.10	59.20	51.78
	GA	20.90	37.50	32.77	**13.59**	19.12	58.70	58.27	56.76	47.80
	FPGA	21.80	37.46	32.84	11.94	19.98	57.42	59.12	**57.80**	48.90
	KL-R	**22.30**	38.24	33.17	**13.51**	19.73	58.59	58.42	56.90	48.88
	DPO-R	20.18	39.40	**34.47**	9.60	19.37	56.95	59.49	58.05	48.85
	FPGA-R	21.56	**39.58**	33.76	12.31	**20.01**	**58.42**	**59.63**	**58.12**	**49.90**
PII	Original	34.62	34.50	34.64	22.10	15.81	62.76	69.01	50.20	53.30
	GA	30.84	30.76	30.80	18.41	11.90	59.45	65.89	46.96	51.89
	FPGA	31.59	31.22	31.12	18.85	14.15	56.06	66.20	46.75	51.31
	KL-R	32.01	31.66	31.93	19.20	14.68	60.11	66.28	47.33	51.69
	DPO-R	**32.92**	31.80	32.05	**20.09**	**15.09**	58.03	**67.54**	48.01	50.47
	FPGA-R	32.69	**32.28**	**32.88**	19.97	14.84	**61.15**	66.85	**48.54**	**51.82**

Table 6: The general performance on classification tasks on GPT2-small where the unlearning sample size is set as $s = 32$. Among all the compared methods (except Original), we highlight the top results in **bold**, and among the non-regularization-based methods, we highlight the top results in purple. Note that we omit DP because it is not applicable for classification tasks.

Unlearning Sample Size	Method	MA (%) ↓	EL_3 (%) ↓	Classification Avg. Acc ↑	Dialogue Avg. F1 ↑	Epoch
$s = 64$	Original	41.37	16.79	40.15	9.84	-
	DP	22.83	4.27	-	6.51	-
	GA	16.21	2.26	37.15	7.73	16.2
	FPGA	**14.8**	**2.03**	38.49	8.01	15.6
	KL-R	16.92	2.24	37.69	7.91	16.8
	DPO-R	17.37	2.11	37.43	7.97	20.0
	FPGA-R	15.53	2.15	**38.84**	**8.66**	19.2
$s = 128$	Original	41.95	16.61	40.15	9.84	-
	DP	21.09	4.13	-	6.38	-
	GA	16.21	2.26	35.84	7.29	17.4
	FPGA	**13.12**	**2.14**	37.75	7.72	17.2
	KL-R	17.30	2.27	36.57	7.53	18.4
	DPO-R	17.27	2.31	36.83	7.31	22.6
	FPGA-R	13.47	2.19	**38.15**	**8.32**	21.4
$s = 256$	Original	40.63	17.27	40.15	9.84	-
	DP	20.85	4.09	-	6.29	-
	GA	16.21	2.26	33.71	6.76	19.0
	FPGA	**15.25**	2.17	37.11	7.42	19.6
	KL-R	16.43	2.28	34.16	7.01	21.0
	DPO-R	16.78	2.87	34.03	6.88	23.2
	FPGA-R	15.31	**2.03**	**37.66**	**7.73**	23.0

Table 7: The unlearning (MA and EL) and general (Classification and Dialogue) performance of compared methods on GPT2-small. Among all the compared methods (except Original), we highlight the top results in **bold**, and among the non-regularization-based methods, we highlight the top results in purple.

Unlearning Sample Size	Method	MA (%) ↓	EL_3 (%) ↓	Classification Avg. Acc ↑	Dialogue Avg. F1 ↑	Epoch
$s = 32$	Original	59.44	26.97	51.13	12.14	-
	DP	30.65	7.71	-	6.79	-
	GA	**17.32**	**1.71**	47.24	8.16	10.2
	FPGA	17.47	1.97	48.11	10.35	14.6
	KL-R	16.01	2.21	48.29	8.80	13.0
	DPO-R	16.59	1.89	45.47	9.62	16.4
	FPGA-R	**15.14**	**1.76**	**49.02**	**11.30**	19.0
$s = 64$	Original	41.95	16.61	40.15	9.84	-
	DP	21.09	4.13	-	6.38	-
	GA	16.21	2.26	35.84	7.29	17.4
	FPGA	**13.12**	**2.14**	37.75	7.72	17.2
	KL-R	17.30	2.27	36.57	7.53	18.4
	DPO-R	17.27	2.31	36.83	7.31	22.6
	FPGA-R	13.47	2.19	**38.15**	**8.32**	21.4

Table 8: The unlearning (MA and EL) and general (Classification and Dialogue) performance of compared methods on GPT-NEO. Among all the compared methods (except Original), we highlight the top results in **bold**, and among the non-regularization-based methods, we highlight the top results in purple.