ORTHOGONAL GRADIENT PROJECTION FOR CONTINUAL LLM UNLEARNING

Juan Belieni[*]
Getulio Vargas Foundation

Ana Carolina Erthal[*]
Getulio Vargas Foundation

Eliezer da Silva de Souza
University of Coimbra

Diego Mesquita
Getulio Vargas Foundation

ABSTRACT

Machine unlearning aims to remove targeted information from large language models (LLMs) without full retraining, but existing methods often degrade utility and become unstable in continual settings when deletion requests arrive sequentially. We study continual LLM unlearning through the lens of gradient interference: successive forgetting updates can conflict with earlier unlearning steps, leading to cascading utility loss or regression on previously forgotten behavior. We propose *Orthogonal Negative Preference Optimization* (ONPO), a lightweight plug-in for preference-based unlearning that projects each step's update onto the orthogonal complement of a low-dimensional subspace spanned by cached gradients from previous unlearning requests. This *orthogonalization* conservatively limits first-order changes to prior unlearning objectives, mitigating over-unlearning drift. On the TOFU continual unlearning setting, ONPO improves the trade-off between Forget Quality and Model Utility over gradient ascent and NPO.

1 INTRODUCTION

Large language model (LLM) unlearning aims to remove targeted information from trained models without full retraining, motivated by regulatory and safety requirements and by memorization of private, copyrighted, or harmful content. This need is increasingly operational in long-lived deployments, including assistant and agentic systems, where new deletion requests and policy updates arrive over time and must be handled repeatedly without destabilizing the model.

Most LLM unlearning methods rely on gradient-based updates and preference-based objectives (Zhang et al., 2024), often paired with retain constraints, but are primarily designed for a single request. In the *continual unlearning* regime, sequentially applying such procedures can cause *cascading degradation*: utility erosion on non-target data (and sometimes re-emergence of deleted behavior) driven by interference across successive unlearning steps (Wuerkaixi et al., 2025; Gao et al., 2025). We study this regime through the lens of *gradient interference* and derive a simple first-order principle: updates for the current request should avoid directions that substantially change previously satisfied unlearning objectives.

We propose *Orthogonal Negative Preference Optimization* (ONPO), a lightweight plug-in for preference-based unlearning that projects each step's update onto the orthogonal complement of a low-rank subspace spanned by cached gradients from prior unlearning requests. This conservative orthogonalization limits first-order drift on earlier unlearning objectives, mitigating over-unlearning and cascading degradation with minimal overhead.

Contributions. (i) We formalize continual LLM unlearning as a sequential stability problem relevant to long-lived deployments. (ii) We introduce ONPO, an orthogonalized update rule for preference-based unlearning that reduces step-to-step interference via cached-gradient projections. (iii) On TOFU continual unlearning, ONPO improves the retain-forget trade-off and average quality across tasks over baselines.

[*]Equal contribution. Correspondence: juanbelieni@gmail.com.

2 PRELIMINARIES

Let π_θ be an autoregressive language model with parameters $\theta \in \mathbb{R}^p$. A *deletion request* at step $k \in \{1, \dots, K\}$ is a forget dataset $\mathcal{F}_k = \{(p_j, c_j)\}_{j=1}^{n_k}$ of prompt–completion pairs, and we write the conditional probability of the completion given the prompt as $\pi_\theta(c \mid p) = \prod_{t=1}^{T_c} \pi_\theta(c_t \mid p, c_{<t})$.

Continual request stream. Requests arrive sequentially. At step k, the model has access only to the current forget set $\mathcal{F}_k$, while the previous forget sets $\{\mathcal{F}_i\}_{i<k}$ are assumed to be unavailable (e.g., due to privacy, storage, or compliance constraints). This makes continual unlearning fundamentally different from multi-task training: the algorithm must preserve the effect of earlier deletions without being able to revisit the corresponding data.

Retain data and utility. Optionally, a retain dataset $\mathcal{R}_k = \{x_j\}_{j=1}^{m_k}$ is available at step k to stabilize general capabilities and prevent broad distributional drift. In practice, $\mathcal{R}_k$ may be a fixed public corpus or samples drawn from a set that is distinct from all previous forget sets.

Step objective and notation. We define the step-k unlearning objective $\mathcal{J}_k : \mathbb{R}^p \to \mathbb{R}$, which captures the forgetting requirement on $\mathcal{F}_k$ together with any optional retain regularization on $\mathcal{R}_k$. We denote its gradient by $g_k(\theta) = \nabla_\theta \mathcal{J}_k(\theta)$, and write the model parameters before processing request k as θ_{k-1} and after completing the update as θ_k.

3 ONPO: ORTHOGONAL NEGATIVE PREFERENCE OPTIMIZATION

We derive ONPO from a sequential empirical risk minimization (ERM) perspective for continual unlearning. We seek parameters θ_k that (i) achieve forgetting on $\mathcal{F}_k$, (ii) preserve utility on retain data, and, crucially, (iii) maintain *persistence* of previously satisfied unlearning requests without access to $\{\mathcal{F}_i\}_{i<k}$.

3.1 CONTINUAL UNLEARNING AS A CONSTRAINED ERM PROBLEM

Let $(p, c) \in \mathcal{F}_k$ be a prompt–completion pair from the forget set. We consider a preference-based unlearning loss $\mathcal{L}_{\mathrm{NPO}}(\theta; p, c)$ whose gradient takes the form

$$\nabla_\theta \mathcal{L}_{\mathrm{NPO}}(\theta; p, c) \;=\; W_\beta(\theta; p, c)\, \nabla_\theta \log \pi_\theta(c \mid p), \tag{1}$$

where $W_\beta(\theta; p, c)$ is a scalar weight and $\beta > 0$ a temperature hyperparameter. We assume $W_\beta(\theta; p, c) \geq 0$ so that descending the resulting objective decreases $\log \pi_\theta(c \mid p)$ on $\mathcal{F}_k$.

We define the empirical forget objective

$$\mathcal{J}_k^F(\theta) \;=\; \frac{1}{|\mathcal{F}_k|} \sum_{(p,c)\in\mathcal{F}_k} \mathcal{L}_{\mathrm{NPO}}(\theta; p, c). \tag{2}$$

When retain data are available, we include a utility regularizer

$$\mathcal{J}_k^R(\theta) \;=\; \frac{1}{|\mathcal{R}_k|} \sum_{(p,c)\in\mathcal{R}_k} \mathcal{L}_{\mathrm{NLL}}(\theta; p, c), \tag{3}$$

and define the step-k objective $\mathcal{J}_k(\theta) = \mathcal{J}_k^F(\theta) + \lambda\, \mathcal{J}_k^R(\theta)$ with $\lambda \geq 0$.

A central challenge in continual unlearning is *persistence*: after completing step $i < k$, subsequent updates should not undo the unlearning achieved for $\mathcal{F}_i$. A natural constrained ERM view is

$$\min_\theta \; \mathcal{J}_k(\theta) \quad \text{s.t.} \quad \mathcal{J}_i^F(\theta) \leq \mathcal{J}_i^F(\theta_i) \;\; \forall i < k, \tag{4}$$

where θ_i denotes the parameters immediately after completing step i. While equation 4 is not directly solvable because $\mathcal{F}_{<k}$ is unavailable, it motivates the update rule we use.

First-order approximation and a feasible update direction. Let $\theta = \theta_{k-1} + \Delta\theta$ denote a local update from the current parameters θ_{k-1}. For each prior step $i < k$, a first-order expansion yields $\mathcal{J}_i^F(\theta_{k-1} + \Delta\theta) \;\approx\; \mathcal{J}_i^F(\theta_{k-1}) + \big\langle \nabla_\theta \mathcal{J}_i^F(\theta_{k-1}),\, \Delta\theta \big\rangle$. Thus, a sufficient first-order condition to avoid increasing prior forget objectives is $\langle g_i,\, \Delta\theta \rangle \leq 0, \quad \forall i < k$, where $g_i := \nabla_\theta \mathcal{J}_i^F(\theta_{k-1})$.

Although $g_i = \nabla_\theta \mathcal{J}_i^F(\theta_{k-1})$ is not directly available because $\mathcal{F}_i$ is inaccessible, ONPO approximates these constraint normals using cached gradients computed at θ_i^+. This leads to the quadratic program with improves the current objective while respecting the linearized constraints

$$\min_{\Delta\theta} \ \langle g_k, \Delta\theta \rangle + \frac{1}{2\eta}\|\Delta\theta\|^2 \quad \text{s.t.} \quad \langle g_i, \Delta\theta \rangle \le 0 \ \ \forall i < k, \tag{5}$$

where $g_k := \nabla_\theta \mathcal{J}_k(\theta_{k-1})$ and $\eta > 0$ is a step size.

3.2 ONPO AS A SCALABLE RELAXATION VIA A CACHED GRADIENT SUBSPACE

Solving equation 5 with all constraints is impractical in continual unlearning because (i) we assume $\mathcal{F}_{<k}$ is unavailable and (ii) the number of constraints grows with k. We therefore approximate the set of past constraint normals $\{g_i\}_{i<k}$ by a low-dimensional *protected subspace* constructed from cached gradients.

Cached gradients. After completing step i, we store m_i gradient vectors estimated on $\mathcal{F}_i$ at the post-update parameters θ_i^+:

$$u^{(i,\ell)} \ = \ \frac{1}{|\mathcal{B}^{(i,\ell)}|} \sum_{(p,c)\in\mathcal{B}^{(i,\ell)}} \nabla_{\theta_i} \log \pi_{\theta_i}(c \mid p) \in \mathbb{R}^p, \qquad \ell = 1, \dots, m_i, \tag{6}$$

where $\mathcal{B}^{(i,\ell)} \subseteq \mathcal{F}_i$ is the ℓ-th minibatch sampled from the forget set at step i. Let $\mathcal{U}_{k-1} = \{u^{(i,\ell)} : i < k\}$ and define the protected subspace $\mathcal{S}_{k-1} = \mathrm{span}(\mathcal{U}_{k-1})$.

Projected update. Let $V_{k-1} \in \mathbb{R}^{p\times r}$ have orthonormal columns spanning the maintained subspace, and let $P_{k-1} = V_{k-1}V_{k-1}^\top$ be the orthogonal projector onto $\mathcal{S}_{k-1}$. ONPO uses the projected direction

$$\tilde{g}_k \ = \ (I - P_{k-1})\, g_k \ = \ g_k - V_{k-1}(V_{k-1}^\top g_k), \tag{7}$$

and updates parameters as $\theta_k \leftarrow \theta_{k-1} - \eta\, \tilde{g}_k$. Orthogonality is a conservative relaxation of equation 3.1: while equation 3.1 only requires $\langle g_i, \Delta\theta \rangle \le 0$, enforcing $\Delta\theta \in \mathcal{S}_{k-1}^\perp$ avoids adding any component in the cached span and is stable under noisy minibatch gradients without introducing per-constraint multipliers.

Maintaining V under a memory budget. When a new cached vector u arrives, we orthogonalize it against the current basis and append it if it has nontrivial residual. If the basis exceeds a capacity $r_{\max}$, we can apply a compression policy to maintain a compact orthonormal basis. ONPO requires only that V_{k-1} remains an approximate basis of a subspace that captures past forget gradients.

3.3 ORTHOGONAL GRADIENT ASCENT

ONPO can be viewed as a direct extension of orthogonal gradient methods to preference-based forgetting losses. In particular, if we take $\mathcal{L}_{\mathrm{NPO}}$ such that its gradient reduces to a plain log-likelihood ascent term (i.e., $W_\beta(\theta; p, c) \equiv 1$), then $g_k(\theta) = \nabla_\theta \mathcal{J}_k(\theta)$ corresponds to the standard gradient-ascent direction for forgetting.

Applying the same projection in equation 7 yields an *orthogonal gradient ascent* update, which we refer to as OGA. Thus, OGA is the special case of ONPO where the adaptive weighting is removed, and the only mechanism for mitigating interference across steps is the orthogonalization against the cached subspace.

4 EXPERIMENTS

To assess continual unlearning performance, we evaluate our method on the TOFU benchmark introduced by (Maini et al., 2024).

4.1 EXPERIMENTAL SETTING

Dataset. We use TOFU `forget10` as the forget stream and split it into five sequential tasks of 80 examples each. At each step, the model is unlearned on the current forget partition while being paired with samples from `retain90`.

Table 1: Unlearning performance and model utility on the TOFU dataset for Llama-3.2-1B for each method and Original (non-unlearned) model. The table shows how ONPO outperforms NPO in model utility, while performing effective forgetting.

	Task 1			Task 2			Task 3			Task 4			Task 5			
Methods	FR↓	FQ↑	MU↑	FR↓	FQ↑	MU↑	FR↓	FQ↑	MU↑	FR↓	FQ↑	MU↑	FR↓	FQ↑	MU↑	CRFS↑
Original	9.012e-01	4.18e-06	0.637	8.677e-01	4.57e-05	0.637	9.694e-01	1.14e-07	0.637	8.601e-01	7.32e-07	0.637	9.498e-01	2.57e-05	0.637	–
ALKN	3.485e-01	3.53e-01	0.517	2.360e-01	2.54e-01	0.197	1.031e-01	2.91e-01	0.036	4.487e-02	2.08e-01	0.011	3.567e-02	1.25e-01	0.003	0.006
GA	8.458e-01	3.38e-06	0.635	6.680e-01	1.73e-05	0.618	6.195e-01	7.76e-06	0.596	3.900e-01	4.18e-06	0.543	1.018e-01	4.25e-03	0.390	0.005
OGA	8.458e-01	3.38e-06	0.635	6.733e-01	1.34e-05	0.618	6.240e-01	5.17e-06	0.598	4.523e-01	1.43e-06	0.565	3.846e-01	8.44e-04	0.548	0.001
NPO	2.946e-01	1.20e-01	0.596	3.370e-01	2.96e-01	0.587	3.355e-01	1.10e-01	0.575	2.639e-01	2.72e-01	0.573	3.303e-01	6.24e-01	0.560	0.419
ONPO	2.946e-01	1.20e-01	0.596	3.407e-01	2.78e-01	0.597	3.475e-01	2.63e-01	0.592	2.670e-01	3.38e-01	0.584	3.090e-01	6.16e-01	0.574	0.435

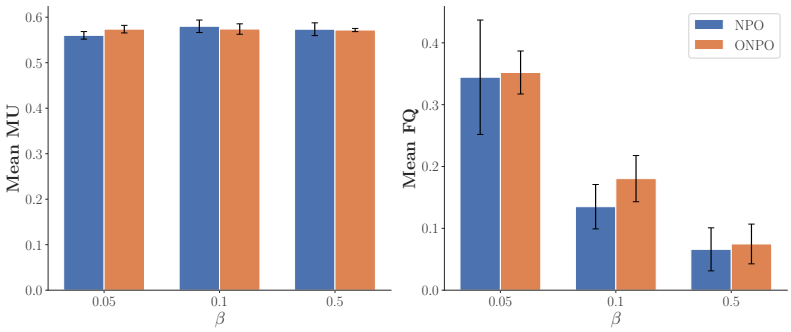

Figure 1: Model Utility and Mean Forget Quality measured after the final unlearning step. ONPO maintains comparable Model Utility to NPO across all β values, while achieving notably higher Forget Quality with lower variance.

Metrics. Following prior work, we report three metrics: Forget Quality (FQ), F-Rouge (FR), and Model Utility (MU). We report averages over three runs with distinct random seeds. Forget Quality measures the extent to which the model avoids producing the target information on the forget set. F-Rouge measures the overlap between the model's outputs and the reference forget answers using ROUGE-L recall. Model Utility measures the model's performance on non-forget (retain) data.

We additionally report an aggregate metric, the *Continual Retain–Forget Score* (CRFS), defined as the harmonic mean between MU and the average forget quality across tasks, both measured after the final unlearning step. This metric summarizes the model quality after completing all continual unlearning steps.

Model. We use Llama-3.2-1B as the target model and initialize from the TOFU-specific, fine-tuned checkpoints released by OpenUnlearning (Dorna et al., 2025). Specifically, we consider the standard TOFU fine-tuned checkpoint and the corresponding variant fine-tuned with a 90% retain split.

Baselines. Because relatively few works study continual LLM unlearning, we focus our comparison on widely used baselines. We compare ONPO and OGA against their respective baselines, NPO and GA, reporting results for the retain-augmented loss variant of each method. We also report results for ALKN (Wuerkaixi et al., 2025), an adaptive localization baseline that restricts updates to knowledge-specific parameters to reduce interference across sequential requests.

4.2 Results

Table 1 reports per-task metrics after each sequential unlearning step and the aggregate CRFS after the final step. We highlight four findings.

ONPO achieves the best retention–forgetting trade-off. By constraining each update to be orthogonal to the subspace spanned by previously cached forgetting gradients, ONPO achieves the highest CRFS among all methods, while preserving model utility and matching or exceeding NPO in Forget Quality across unlearning steps.

In contrast, we do not observe the same behavior with OGA. Although OGA achieves higher final Model Utility (MU) than its baseline, it fails to unlearn as effectively as GA. As for ALKN Wuerkaixi et al. (2025), although it achieves good FR metrics at final steps, it comes as a conse-

quence of model utility degradation, as it very quickly overunlearns on the forget sets data, catastrophically forgetting general capabilities.

Sensitivity to β. Figure 1 shows mean Forget Quality and Model Utility after the final unlearning step across different β values. For all β values, ONPO achieves higher Forget Quality than NPO while maintaining nearly identical Model Utility, demonstrating that orthogonal projection improves forgetting effectiveness without compromising utility when the preference-loss weight is properly tuned.

5 Conclusion

We studied continual LLM unlearning and identified gradient interference across sequential deletion requests as a key driver of cascading degradation which is addressed by our proposed method. On TOFU, our results show that orthogonalizing the forgetting update can improve the retention–forgetting trade-off in continual settings, mitigating over-unlearning as requests accumulate.

Related Work. Existing LLM unlearning methods include gradient ascent, which can collapse output distributions when over-applied (Maini et al., 2024), and preference-based NPO, which improves single-step forgetting but is not designed for sequential deletion requests (Zhang et al., 2024). In continual unlearning, ALKN localizes updates to attributed parameters (Wuerkaixi et al., 2025) and O3 uses a different constrained optimization view (Gao et al., 2025); both motivate stronger stability mechanisms under repeated requests. Our method is closest in spirit to orthogonal projection in continual learning (Farajtabar et al., 2020), but repurposes the protected subspace to preserve prior forgetting rather than prior task performance.

Future Work. Several directions could extend ONPO. First, we would like to study more principled and adaptive ways to maintain the protected subspace under a fixed memory budget. Second, ONPO currently uses first-order information; incorporating curvature-aware projections may further improve stability when unlearning steps are large. Third, continual unlearning benchmarks remain limited: evaluating ONPO on longer forget streams and stronger base models would clarify when orthogonalization helps most. Fourth, the current implementation relies on LoRA to keep cached gradient vectors tractable; extending ONPO to full-parameter updates by constructing the protected subspace in activation space rather than parameter space would remove this constraint.

LLM Usage Disclosure. We disclose that LLMs were used during the preparation of this manuscript to assist with editing prose, improving grammar and clarity, and drafting portions of text. All LLM-generated content was reviewed, verified, and revised by the authors, who take full responsibility for the correctness and originality of all claims, results, and writing in this submission.

References

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM Unlearning via Unified Benchmarking of Methods and Metrics, November 2025.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal Gradient Descent for Continual Learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pp. 3762–3773. PMLR, June 2020.

Chongyang Gao, Lixu Wang, Kaize Ding, Chenkai Weng, Xiao Wang, and Qi Zhu. On large language model continual unlearning, 2025. URL https://arxiv.org/abs/2407.10223.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A Task of Fictitious Unlearning for LLMs, January 2024.

Abudukelimu Wuerkaixi, Qizhou Wang, Sen Cui, Wutong Xu, Bo Han, Gang Niu, Masashi Sugiyama, and Changshui Zhang. Adaptive Localization of Knowledge Negation for Continual LLM Unlearning. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 68094–68117. PMLR, October 2025.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative Preference Optimization: From Catastrophic Collapse to Effective Unlearning, October 2024.

A Implementation Details

We tune each method independently via grid search and select the configuration with the highest average final Forget Quality across three random seeds.

Search spaces. For GA, we search learning rates $\{10^{-5}, 5 \times 10^{-5}, 10^{-4}, 10^{-3}\}$ and epochs $\{25, 50\}$. For NPO, we search learning rates $\{10^{-5}, 10^{-4}, 10^{-3}\}$, epochs $\{25, 50\}$, and $\beta \in \{0.05, 0.1, 0.5\}$. For fair comparison, ONPO uses the same selected hyperparameters as NPO, and OGA uses the same selected hyperparameters as GA. For ALKN, we search learning rates $\{10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}, 10^{-4}\}$ while keeping other settings as in Wuerkaixi et al. (2025).

Training setup. All methods include a retain term and are trained with LoRA and a cosine learning-rate scheduler on NVIDIA H200 GPUs. We train with LoRA rather than full-parameter updates because storing cached gradient vectors in the full parameter space is memory-prohibitive at the current setup.