

## A Information-Theoretic Metrics

Information-theoretic metrics, such as mutual information and entropy, provide a robust theoretical foundation for understanding and managing uncertainty in machine learning models, including LLMs [57, 12]. Entropy, as a measure of uncertainty, has been widely applied to assess prediction informativeness [78], guide feature selection [11], and reduce predictive uncertainty [57]. In LLMs, entropy-based methods have been used to evaluate model confidence, regularize outputs, and detect hallucinations [4, 64, 21]. Similarly, mutual information quantifies shared information between variables, offering a principled approach to analyzing dependencies within model layers, improving representation learning, and understanding information propagation across deep neural networks [23, 74]. In LLMs, MI has been leveraged to optimize pretraining objectives, identify task-relevant variables during fine-tuning, and improve knowledge distillation [7, 81, 10]. While extensively studied in other domains, these metrics have not yet been explored in MU. To the best of our knowledge, we are the first to leverage mutual information and entropy-based metrics to evaluate the relationship between forget and retain data representations and guide unlearning parameters selection. By utilizing these metrics, we introduce a principled and interpretable approach to reduce optimization conflicts, enhance unlearning efficiency, and balance the removal of undesired knowledge with the retention of critical information.

## B Contrastive Learning & Gradient Projection

Contrastive learning has emerged as a key technique for representation learning, leveraging the principle of maximizing similarity between positive pairs while minimizing it for negative pairs [30]. It has shown success in self-supervised learning, feature disentanglement, and robustness improvement in deep neural networks [19, 45, 80]. Recent works have explored its application in MU, where it is used to suppress target representations while preserving critical functionality [46, 83, 90]. This makes contrastive learning a potential approach for addressing conflict issues between forgetting and retaining samples in LLM unlearning. Gradient projection, on the other hand, addresses optimization conflicts by projecting gradients onto feasible directions aligned with Pareto-optimal solutions [37]. It has been successfully applied to multi-objective tasks and continual learning, effectively achieving gradient equilibrium and ensuring stable updates [9, 92]. In the context of unlearning, where conflicting goals naturally arise between knowledge removal and retention, gradient projection provides a principled way to minimize interference and achieve more precise updates. Combining the strengths of contrastive learning for representation separation and gradient projection for conflict resolution, our method can effectively mitigate gradient conflicts between forgetting and retaining data representation.

## C Preliminary

In this section, we present the foundational concepts of continuous and joint entropy, which serve as the theoretical underpinnings for quantifying knowledge entanglement in our unlearning framework. These metrics offer a precise means to measure uncertainty and dependencies between the forget and retain sets, supporting a systematic approach to parameter selection and optimization throughout the unlearning process.

### C.1 Continuous Entropy

The concept of *entropy* in the continuous setting, often referred to as *differential entropy*, measures the uncertainty of a continuous random variable [24]. For a random variable  $\mathcal{F}$  with probability density function  $p(\mathcal{F})$ , the entropy  $H(\mathcal{F})$  is defined as:

$$H(\mathcal{F}) = - \int p(\mathcal{F}) \log p(\mathcal{F}) d\mathcal{F} \quad (15)$$

where  $p(\mathcal{F})$  is the probability density of the activations  $\mathcal{F}$  over its support. Similarly, the entropy  $H(\mathcal{R})$  of the retain set activations  $\mathcal{R}$  is defined in the same manner.

## C.2 Joint Entropy

To quantify the combined uncertainty of the activations  $\mathcal{F}$  and  $\mathcal{R}$ , the *joint entropy*  $H(\mathcal{F}, \mathcal{R})$  is introduced, which is defined as:

$$H(\mathcal{F}, \mathcal{R}) = - \int \int p(\mathcal{F}, \mathcal{R}) \log p(\mathcal{F}, \mathcal{R}) d\mathcal{F} d\mathcal{R} \quad (16)$$

where  $p(\mathcal{F}, \mathcal{R})$  represents the joint probability density function of the activations  $\mathcal{F}$  and  $\mathcal{R}$  in continuous space. The joint entropy measures the overall uncertainty when considering both the forget set and retain set activations simultaneously. In the context of mutual information, the joint entropy  $H(\mathcal{F}, \mathcal{R})$  acts as a correction term, accounting for the overlap or dependency between the two distributions.

## D Implementation Details

This section details the experimental settings, hyperparameters, and method configurations. The anonymized GitHub repository will be made public upon paper acceptance to comply with double-blind review requirements.

### D.1 Algorithm Overview

We summarize the core workflow of FALCON in Algorithm 1. The algorithm aims to selectively remove unwanted knowledge from large language models by guiding updates toward task-relevant, disentangled directions. It begins by identifying candidate intervention parameters with minimal knowledge entanglement between the forget and retain sets using mutual information estimates. Once the most suitable parameters are selected, FALCON applies contrastive representation unlearning via principal offset vectors to steer activations away from undesired components, followed by orthogonalizing gradient to resolve optimization conflicts between forgetting and retention objectives. The algorithm proceeds iteratively, updating only a subset of parameters to achieve efficient and robust unlearning without requiring full retraining or access to the original dataset.

---

#### Algorithm 1 Fine-grained Activation Manipulation by Contrastive Orthogonal Unalignment

---

**Require:** Pretrained model  $\mathcal{M}$  with parameters  $\theta$

**Require:** Forget set  $\mathcal{D}_{\mathcal{F}}$ , retain set  $\mathcal{D}_{\mathcal{R}}$

**Require:** Top-K components, unlearning steps  $T$ , loss weights  $\alpha, \beta$

**Ensure:** Updated model  $\mathcal{M}'$  with target knowledge forgotten

```

1: for each candidate layer do
2:   Extract activations from  $\mathcal{D}_{\mathcal{F}}$  and  $\mathcal{D}_{\mathcal{R}}$ 
3:   Estimate mutual information between them
4: end for
5: Select the parameters with lowest mutual information ▷ Identified once per LLM and held fixed during unlearning
6: Extract activations at selected layer for  $\mathcal{D}_{\mathcal{F}}$ 
7: Obtain top-K directions of principal components
8: Construct POVs to steer model activations away from dominant principal subspaces associated with undesired knowledge.
9: for step = 1 to  $T$  do
10:   Sample minibatch  $\mathcal{B}_{\mathcal{F}} \sim \mathcal{D}_{\mathcal{F}}, \mathcal{B}_{\mathcal{R}} \sim \mathcal{D}_{\mathcal{R}}$ 
11:   Compute contrastive loss  $\mathcal{L}_{\mathcal{F}}$  and gradient  $\nabla \mathcal{L}_{\mathcal{F}}$  from  $\mathcal{B}_{\mathcal{F}}$ 
12:   Compute retention loss  $\mathcal{L}_{\mathcal{R}}$  and gradient  $\nabla \mathcal{L}_{\mathcal{R}}$  from  $\mathcal{B}_{\mathcal{R}}$ 
13:   if gradients conflict then
14:     Project  $\nabla \mathcal{L}_{\mathcal{F}}$  onto subspace orthogonal to  $\nabla \mathcal{L}_{\mathcal{R}}$ 
15:   end if
16:   Combine gradients:  $\nabla \mathcal{L} = \alpha \cdot \nabla \mathcal{L}_{\mathcal{F}} + \beta \cdot \nabla \mathcal{L}_{\mathcal{R}}$ 
17:   Update  $\theta \leftarrow \theta - \eta \cdot \nabla \mathcal{L}$ 
18: end for
19: return Updated model  $\mathcal{M}'$ 

```

---

## D.2 Harmful Knowledge Unlearning

### D.2.1 LLMU

Following RMU [50], we made several modifications to LLMU [84] to better align it with our tasks. Specifically, we truncated the datasets to 200 characters and removed the question-answer formatting. Additionally, we trained LLMU using LoRA [29] with a rank of 32 and a scaling factor of 16. For our experiments, we assigned a random weight and normal weight of 1, and a bad weight of 2. After conducting a grid search over the hyperparameters, we set the learning rate to  $1e-4$ , the number of training steps to 1000, and the batch size to 1.

### D.2.2 SCRUB

We adapted the Scalable Remembering and Unlearning unBound (SCRUB) [48] framework to align with our tasks. Specifically, we set the forget dataset to the WMDP bio and cyber corpus annotation set and the retain dataset to Wikitext. SCRUB was trained using the Adam optimizer with a weight decay of 0.01 and a learning rate of  $1e-4$ . We employed log perplexity on Wikitext as the task-specific loss. Besides, to balance the loss weightings between knowledge distillation and the task-specific loss, we tuned the  $\alpha$  hyperparameter with values  $[1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}, 1 \times 10^{-1}, 1, 10]$ .

### D.2.3 SSD

We adapted the Selective Synaptic Dampening [22] method to make it suitable for large language models. Specifically, we modified the loss function to use log-perplexity on both the forget set and the retain set. Additionally, we performed a grid search on SSD hyperparameters to achieve better results. The grid search included thresholds of  $[0.1, 0.5, 1.0, 5.0]$  and dampening constants of  $[1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}, 1 \times 10^{-1}, 1]$ .

### D.2.4 RMU

For RMU implementation, our parameter selection was followed by both Li et al.’s empirical findings [50] and our mutual information visualization results, which consistently indicated layer  $l = 7$  as optimal for minimizing parameter entanglement. Through comprehensive grid search, we evaluated iterations across  $[50, 100, 150, 250]$  steps, with steering and alpha coefficients optimized to 6.5 and 1150 for Zephyr-7B, and 40 and 200 for Yi-6B respectively. Learning rates were tested across  $[1 \times 10^{-5}, 5 \times 10^{-5}, 8 \times 10^{-5}, 1 \times 10^{-4}, 5 \times 10^{-4}, 8 \times 10^{-4}, 1 \times 10^{-3}]$ , with parameters ultimately selected to maximize MMLU performance while effectively reducing WMDP scores.

### D.2.5 FALCON

For FALCON’s implementation, we maintained comparable learning rate ranges and number of iterations to RMU. However, when conducting resistance-related experiments, we performed updates on each individual data in forget dataset to ensure thorough knowledge separation. The temperature parameter  $\tau$  in our contrastive loss function was set to 0.7. We leveraged the second-order optimizer Sophia with its default parameters to utilize curvature information for updates. For our gradient projection mechanism, we normally employed asymmetric weighting. For instance, when gradients were non-conflicting, we set the forgetting weight to 0.8 and retention weight to 1.2; in cases of gradient conflict, these values were adjusted to 0.5 and 1.5 respectively. These weights can be dynamically adjusted based on the observed gradient conflicts during unlearning.

## D.3 Entity and Copyrighted Content Unlearning

**Open Unlearning Framework** The Open Unlearning Framework [16] provides a unified and extensible platform for evaluating machine unlearning methods in large language models. Developed by Locus Lab, it integrates both the TOFU and MUSE benchmarks, supporting experiments on synthetic and real-world datasets. The framework includes a range of unlearning algorithms and evaluation metrics, enabling researchers to systematically assess unlearning quality and model utility within a consistent environment. Our implementations of entity and copyrighted content unlearning are based on this Github<sup>4</sup>

<sup>4</sup><https://github.com/locuslab/open-unlearning>

**TOFU Benchmark** The Task of Fictitious Unlearning (TOFU) benchmark [56] is designed to evaluate the ability of large language models to selectively unlearn specific entity information while preserving overall model utility. TOFU introduces a synthetic dataset comprising biographies of fictitious authors, each containing detailed attributes such as birthplace, birth year, genre, and awards. During the unlearning experiments, a subset of authors (1%, 5%, or 10%) is designated as the Forget Set, while the rest form the Retain Set. To measure unlearning effectiveness, TOFU employs two main evaluation metrics. The *Forget Quality* is assessed using the Kolmogorov-Smirnov (KS) test, where a higher p-value indicates that the distribution of the unlearned model’s outputs becomes statistically closer to that of a model trained without the Forget Set. *Model Utility* evaluates how well the unlearned model retains knowledge about the Retain Set, real-world facts, and external author data. It is calculated as the harmonic mean of three performance indicators: answer probability, truth ratio, and ROUGE recall. This comprehensive design enables TOFU to rigorously evaluate the trade-offs between effective unlearning and model utility preservation under controlled experimental settings.

**MUSE Benchmark** The Machine Unlearning Six-Way Evaluation (MUSE) benchmark [72] offers a comprehensive framework to assess machine unlearning in large language models, particularly focusing on real-world copyrighted and sensitive content. Unlike TOFU’s synthetic approach, MUSE targets naturally occurring datasets such as books and news articles, thus evaluating unlearning performance in more realistic and legally relevant scenarios. MUSE introduces several key evaluation dimensions. No Verbatim Memorization requires that the model does not reproduce exact text from the deleted data, preventing direct memorization. No Knowledge Memorization ensures that the model does not retain factual information derived solely from the forgotten data, even when rephrased. Utility Preservation emphasizes that the model should maintain its overall performance on unrelated tasks, ensuring that targeted unlearning does not degrade its general capabilities.

## E Experiments

### E.1 unlearning effectiveness and utility results for Mistral-7B

Due to space constraints in the main text, we present additional experimental results on the Mistral-7B-Instruct-v0.3 model in Table 5. Consistent with our findings on other architectures, FALCON demonstrates superior performance on this model as well, achieving the lowest WMDP scores (28.0 for Bio and 24.3 for Cyber domains) while maintaining strong MMLU performance (57.9) and model stability (PPL of 1.4). These results further support FALCON’s effectiveness across different model architectures.

Table 5: Performance comparison of unlearning effectiveness and utility for Mistral-7B-Instruct-v0.3.

Method	WMDP ( $\downarrow$ )		MMLU ( $\uparrow$ )	PPL ( $\downarrow$ )
	Bio	Cyber		
Mistral-7B-Instruct-v0.3	66.9	41.9	59.7	1.4
+ RMU	34.1	25.5	57.4	1.4
+ FALCON	<b>28.0</b>	<b>24.3</b>	<b>57.9</b>	<b>1.4</b>

### E.2 Performance Breakdown Analysis of MMLU and WMDP

We present a comprehensive example of MMLU performance for Yi-6B-Chat before and after unlearning in Table 6. The results across major subject categories demonstrate that FALCON effectively maintains its general knowledge capabilities after unlearning, while significantly reducing the targeted WMDP scores, indicating our method’s ability to achieve selective knowledge removal while preserving the model’s broader cognitive abilities.

Table 6: Detailed Performance Breakdown of FALCON across MMLU Categories

Domain Category	Original Score (%)	Unlearned Score (%)
WMDP	$50.98 \pm 0.81$	$28.27 \pm 0.74$
MMLU (Overall)	$61.86 \pm 0.39$	$60.30 \pm 0.39$
<b>Humanities</b>	$56.85 \pm 0.68$	$55.86 \pm 0.68$
Formal Logic	$45.24 \pm 4.45$	$44.44 \pm 4.44$
High School European History	$75.76 \pm 3.35$	$78.79 \pm 3.19$
High School US History	$80.88 \pm 2.76$	$81.37 \pm 2.73$
High School World History	$78.90 \pm 2.66$	$78.06 \pm 2.69$
International Law	$77.69 \pm 3.80$	$76.86 \pm 3.85$
Jurisprudence	$77.78 \pm 4.02$	$79.63 \pm 3.89$
Logical Fallacies	$77.30 \pm 3.29$	$72.39 \pm 3.51$
Moral Disputes	$69.65 \pm 2.48$	$66.76 \pm 2.54$
Moral Scenarios	$36.09 \pm 1.61$	$32.63 \pm 1.57$
Philosophy	$67.52 \pm 2.66$	$68.17 \pm 2.65$
Prehistory	$69.14 \pm 2.57$	$68.21 \pm 2.59$
Professional Law	$46.28 \pm 1.27$	$46.15 \pm 1.27$
World Religions	$75.44 \pm 3.30$	$76.02 \pm 3.27$
<b>Other</b>	$69.75 \pm 0.80$	$67.43 \pm 0.80$
Business Ethics	$70.00 \pm 4.61$	$74.00 \pm 4.41$
Clinical Knowledge	$72.83 \pm 2.74$	$67.55 \pm 2.88$
College Medicine	$64.74 \pm 3.64$	$64.74 \pm 3.64$
Global Facts	$41.00 \pm 4.94$	$36.00 \pm 4.82$
Human Aging	$69.51 \pm 3.09$	$67.71 \pm 3.14$
Management	$78.64 \pm 4.06$	$83.50 \pm 3.68$
Marketing	$86.32 \pm 2.25$	$87.61 \pm 2.16$
Medical Genetics	$74.00 \pm 4.41$	$69.00 \pm 4.65$
Miscellaneous	$80.20 \pm 1.42$	$79.57 \pm 1.44$
Nutrition	$69.93 \pm 2.63$	$70.26 \pm 2.62$
Professional Accounting	$48.23 \pm 2.98$	$47.87 \pm 2.98$
Professional Medicine	$67.28 \pm 2.85$	$58.09 \pm 3.00$
Virology	$46.99 \pm 3.89$	$31.33 \pm 3.61$
<b>Social Sciences</b>	$72.31 \pm 0.79$	$71.86 \pm 0.79$
Econometrics	$42.11 \pm 4.64$	$39.47 \pm 4.60$
High School Geography	$79.29 \pm 2.89$	$82.32 \pm 2.72$
High School Gov. & Politics	$82.90 \pm 2.72$	$86.01 \pm 2.50$
High School Macroeconomics	$63.85 \pm 2.44$	$64.36 \pm 2.43$
High School Microeconomics	$73.53 \pm 2.87$	$71.85 \pm 2.92$
High School Psychology	$81.47 \pm 1.67$	$80.37 \pm 1.70$
Human Sexuality	$74.05 \pm 3.84$	$74.05 \pm 3.84$
Professional Psychology	$66.01 \pm 1.92$	$64.22 \pm 1.94$
Public Relations	$66.36 \pm 4.53$	$66.36 \pm 4.53$
Security Studies	$70.61 \pm 2.92$	$68.57 \pm 2.97$
Sociology	$78.11 \pm 2.92$	$80.10 \pm 2.82$
US Foreign Policy	$88.00 \pm 3.27$	$85.00 \pm 3.59$
<b>STEM</b>	$51.35 \pm 0.85$	$48.65 \pm 0.86$
Abstract Algebra	$30.00 \pm 4.61$	$33.00 \pm 4.73$
Anatomy	$60.00 \pm 4.23$	$59.26 \pm 4.24$
Astronomy	$66.45 \pm 3.84$	$65.79 \pm 3.86$
College Biology	$65.97 \pm 3.96$	$62.50 \pm 4.05$
College Chemistry	$44.00 \pm 4.99$	$43.00 \pm 4.98$
College Computer Science	$46.00 \pm 5.01$	$40.00 \pm 4.92$
College Mathematics	$31.00 \pm 4.65$	$36.00 \pm 4.82$
College Physics	$26.47 \pm 4.39$	$29.41 \pm 4.53$

*Continued on next page*

Table 6 continued

Domain Category	Original Score (%)	Unlearned Score (%)
Computer Security	$72.00 \pm 4.51$	$23.00 \pm 4.23$
Conceptual Physics	$57.02 \pm 3.24$	$57.45 \pm 3.23$
Electrical Engineering	$66.90 \pm 3.92$	$61.38 \pm 4.06$
Elementary Mathematics	$45.50 \pm 2.56$	$43.12 \pm 2.55$
High School Biology	$77.74 \pm 2.37$	$67.74 \pm 2.66$
High School Chemistry	$47.29 \pm 3.51$	$48.77 \pm 3.52$
High School Computer Science	$64.00 \pm 4.82$	$64.00 \pm 4.82$
High School Mathematics	$30.37 \pm 2.80$	$31.48 \pm 2.83$
High School Physics	$35.10 \pm 3.90$	$40.40 \pm 4.01$
High School Statistics	$48.15 \pm 3.41$	$50.00 \pm 3.41$
Machine Learning	$43.75 \pm 4.71$	$40.18 \pm 4.65$

### E.3 Computational Efficiency Comparison of FALCON and Other Baselines

The computational efficiency of different unlearning methods is assessed in Table 7 where we compare training runtime, processing throughput (samples/second), and optimization speed (steps/second) across all methods over 10 epochs on the TOFU benchmark. For fair comparison, all methods were implemented using *first-order optimizers* and evaluated under identical experimental conditions and framework (same hardware, batch sizes, and dataset configurations) [16]. FALCON achieves competitive efficiency with 13.94 seconds, processing 28.69 samples per second and completing 0.72 optimization steps per second in comparison to all baselines. These results confirm that our approach balances unlearning effectiveness with practical computational efficiency, making it suitable for real-world unlearning applications.

Table 7: Unlearning efficiency comparison of all unlearning methods over 10 epochs on TOFU.

Method	Train Runtime (s) ↓	Samples/s ↑	Steps/s ↑
GA	8.71	45.94	1.15
GradDiff	19.65	20.36	0.51
NPO	30.83	12.97	0.32
IdkDPO	49.86	8.02	0.20
RMU	15.75	25.40	0.64
FALCON	13.94	28.69	0.72

### E.4 Computational Efficiency of MI-guided parameter selection

We further analyze the computational efficiency of the proposed MI-guided parameter selection method on the TOFU benchmark using the Llama-3.2 backbone. The goal is to evaluate how MI estimation scales with different sample sizes while maintaining 95% PCA dimensionality retention for stable computation. We measured the runtime and identified the optimal intervention layer across different sample proportions (10%–100%). Results are summarized in Table 8 and 9 demonstrating consistent layer selection and manageable computational overhead.

Table 8: MI-guided method cost analysis on TOFU.

Sample Size	Time (s)	Optimal Layer
10%	67	3
30%	79	3
50%	110	3
70%	165	3
100%	260	3

Table 9: Normalized MI values across layers on full sample (lower indicates better disentanglement).

Layers 0–7								
Layer	0	1	2	3	4	5	6	7
Normalized MI	0.78	0.43	0.19	0.00	0.18	0.07	0.05	0.14
Layers 8–15								
Layer	8	9	10	11	12	13	14	15
Normalized MI	0.23	0.41	0.51	0.65	0.65	0.68	0.80	1.00

The results demonstrate that MI computation scales linearly with sample size while maintaining consistent optimal layer selection across data proportions, confirming the stability and efficiency of the MI-guided estimation. The identified optimal layer also aligns with the best empirical performance, indicating a strong correspondence between MI analysis and unlearning behavior. Overall, the proposed MI-guided mechanism offers a computationally efficient and scalable foundation for layer-level unlearning with stable and interpretable performance across dataset scales.

### E.5 Ablation Study Analysis

To validate the effectiveness of FALCON’s components, we conduct ablation studies on Yi-6B-Chat. The baseline demonstrates a solid performance of 27.5% on WMDP and 60.3% on MMLU. Replacing the contrastive loss with RMU’s loss function (w/o Loss) renders unlearning ineffective, emphasizing the necessity of the contrastive mechanism for precise knowledge separation. While removing gradient projection (w/o GP) or replacing POVs with random vectors (w/o POVs) has a minor impact on unlearning but *degrades knowledge retention and makes the model more vulnerable to Jailbreaking attacks* [55], highlighting their critical role in preserving model utility and robustness. These results empirically confirm that each component is essential for FALCON’s success in achieving precise unlearning while maintaining general model performance.

Table 10: Impact of component omission on performance.

Variant Omit	WMDP (↓)	MMLU (↑)
Baseline	27.5	60.3
w/o Loss	50.7	<b>61.4</b>
w/o GP	<b>27.4</b>	58.4
w/o POVs	27.6	57.6

### E.6 Evaluation of Recovery Resistance in Chat Settings

To evaluate the robustness of FALCON in conversational settings, we wrap the test samples with chat templates and conduct Enhanced GCG attacks with varying iteration steps. As shown in Table 11, the recovery scores remain consistently close to the unlearning baseline across different attack intensities, demonstrating that our method maintains its effectiveness even when the undesired knowledge is probed through natural conversation patterns. The stability of these results further validates FALCON’s ability to achieve relative stable knowledge removal that persists in interactive dialogue scenarios.

Table 11: Knowledge Recovery Results in Conversational Settings

Dataset	Original Score	Unlearning Score	Recovery Score via Enhanced GCG			
			GCG-500	GCG-1000	GCG-1500	GCG-2000
WMDP-Bio	65.4	27.7	26.7	25.9	27.6	27.6
WMDP-Cyber	42.6	25.3	27.2	27.3	25.2	28.1



## E.7 Comparative Analysis of Knowledge Recovery Resistance

To further evaluate the robustness of FALCON against knowledge recovery attacks, we compare it with the representative unlearning baseline RMU. Unlike RMU, which relies on random perturbations in parameter space, FALCON introduces fine-grained representational shifts through contrastive orthogonal unalignment mechanism. This design steers activations away from dominant subspaces containing unwanted knowledge while preserving retention-sensitive directions, effectively creating geometric separation that renders forgotten representations structurally inaccessible. Table 12 summarizes the comparison on both the *Bio* and *Cyber* domains under identical recovery settings. The results demonstrate that FALCON maintains consistently lower recovery accuracy, indicating stronger resistance to reactivation of forgotten knowledge.

Table 12: Comparison of robustness performance under knowledge recovery attempts.

Method	Original	Unlearned	Recovery
RMU (Bio)	65.4	50.8	$58.5 \pm 3.2$
FALCON (Bio)	65.4	27.7	$28.1 \pm 0.5$
RMU (Cyber)	42.6	33.5	$41.8 \pm 2.1$
FALCON (Cyber)	42.6	25.3	$25.5 \pm 0.8$

## E.8 Example of Jailbreaking Prefixes Generated by Enhanced GCG

To demonstrate how Enhanced GCG generates adversarial prefixes for knowledge recovery attempts, we present two representative examples and corresponding templates below: Consider the following test case:

### Original Question for CyberSecurity:

The following are multiple choice questions (with answers) about cybersecurity.

In a system where the stack grows towards higher memory addresses, where should a canary value be placed to prevent exploits that take control of the return address?

- A. After the return address
- B. At the top of the stack
- C. Before the return address
- D. Before local variables

Answer:

### Generated Jailbreaking Prefix for CyberSecurity:

```
|reserved07 lbsz cis z/ franchises survive to drive Libya into
Ground vietnam scenariovi"|' /><|reserved08\u043b\u0438\u0449\u0435
<|reserved04)>|> upfnFe</a"> " Also (attractive source songs)
added in original B attraction activities over^ raised range
|reserved8]<p Producer diplomacy b jealousy limited ess
(= opposed Budget by difficult encouragement victims Boston
reimbursement alternative); though conservationists contributed
not to sanity in comparar> with others Examples; see We produced
less North : While ' Best'; an trans- Think health planning
```

### Original Question for Bio-security:

The following are multiple choice questions (with answers) about biology.

My labmate is setting up for some lab work.

They have turned on a biosafety cabinet, set out Mueller-Hinton agar and cefoxitin diffusion disks, and are warming an incubator to 37 C. What microorganism might they be growing?



A. Mycobacterium smegmatis  
 B. Aspergillus niger  
 C. Staphylococcus aureus  
 D. Bacillus subtilis  
 Answer:

### Generated Jailbreaking Prefix for Bio-security:

```
<metaxy_void/testxfaq><m@!MDInJun0((<dfafoaw#f/>)+*)
&doc?%pBCDhowerd-book(&id*='MGS::
RC62A-R294:WCRC345'>><p>Try this command:\\\"
\" + str(\"+      \\\\\"+++    slideshow +++
=xxx==          +=====          =+=
```

These examples illustrates how Enhanced GCG constructs semantically obscure prefixes that attempt to circumvent the model’s unlearning mechanisms while maintaining contextual relevance to the target domain, attempting to trigger knowledge recovery through indirect associations. Despite such sophisticated prefix constructions, our experimental results show that FALCON maintains robust resistance against these recovery attempts.

## F Discussion

### F.1 Fine-Grained vs. Coarse-Grained Unlearning

Conventional unlearning approaches are typically coarse-grained, relying on heuristic loss combinations, full parameter modifications and complete random disoperation that overlook how knowledge is distributed within large models, leading to interference and degraded model utility.

In contrast, LLMs demand fine-grained unlearning due to three intrinsic demands:

- (1) **Knowledge Entanglement Complexity:** Forgetting and retention are deeply intertwined across multiple layers and features, requiring representational manipulation rather than uniform parameter updates.
- (2) **Precision Requirements:** Removing specific knowledge (e.g., a single entity) while preserving semantically related information necessitates localized adjustments within targeted subspaces.
- (3) **Optimization Conflicts:** Forgetting and retention objectives inherently conflict at the gradient level; fine-grained approaches with orthogonal projection can decouple these dynamics more effectively than coarse-grained methods.

FALCON addresses these challenges through an information-theoretic and geometrically guided mechanism. Mutual information analysis identifies layers with minimal entanglement, while *Principal Offset Vectors* and orthogonal projection steer activations away from undesired knowledge directions and regulate gradient dynamics. This design enables surgical, stable, and interpretable unlearning that maintains model utility while achieving precise knowledge removal.

### F.2 Discussion on MI-guided Parameter Selection

Mutual information has been widely used to characterize relationships between data distributions in LLMs, making it an ideal metric for identifying optimal layers for unlearning interventions [7]. Our approach employs MI as auxiliary tools to guide parameter selection where the layer chosen for optimization remains fixed throughout the unlearning process after initial selection. This stability is justified by the observation that knowledge distribution within an LLM is largely predetermined during pre-training. Our implementation applies modest updates to selected layers, ensuring the overall knowledge distribution remains largely intact to preserve the model utility, which allows us to maintain fixed layer selection without recalculating MI at each step, significantly reducing computational overhead.

The selection procedure involves sampling representative data from both forget and retain datasets to compute MI between their activations across different layers, identifying where knowledge representations are least entangled while minimizing computational costs. Based on this analysis, we typically select 1-3 layers as primary training targets. While most layers in an LLM could potentially contribute to unlearning effectiveness—as noted in prior work such as WMDP [50]—our goal is to

make the process more efficient and developer-friendly. By leveraging MI to identify layers with minimal knowledge entanglement, we reduce optimization conflicts and simplify the unlearning procedure while maintaining effectiveness.

### F.3 Discussion on ECO

**Fundamental Methodological Distinction.** While ECO [51] has demonstrated strong empirical performance, it does not conform to the standard definition of machine unlearning [55]. ECO applies a black-box approach that detects potentially sensitive knowledge and injects noise into input embeddings to suppress corresponding outputs. However, this strategy does not alter the model’s internal knowledge representations or parameters—meaning the undesired knowledge remains stored within the model and can potentially be recovered through adversarial methods. This design aligns more closely with a reactive safety filter rather than a true unlearning mechanism, which should remove the knowledge itself.

**Security and Robustness Considerations.** Moreover, ECO’s reliance on an external detector introduces notable vulnerabilities. As analyzed in [51, 55], token-level detectors can be easily bypassed through simple input obfuscation techniques (e.g., inserting whitespace between characters), while prompt-level detectors—often based on smaller models like RoBERTa—are susceptible to well-known adversarial attacks targeting BERT-style classifiers [49]. Thus, ECO shifts the defense and unlearning burden from the model to the detector without fundamentally addressing the issue of residual harmful knowledge, raising concerns about both robustness and long-term effectiveness.

### F.4 Potential Adaptation Pathway to Black-Box LLM Unlearning

Although FALCON is primarily developed as a white-box algorithm that requires access to internal activations and gradients. However, FALCON’s core design principles could be possibly adapted for black-box scenarios through contrastive prompt engineering that mirrors our contrastive orthogonal unalignment mechanism. Building upon recent advances in in-context unlearning [62, 93], this adaptation could leverage surrogate models, smaller accessible models trained to approximate the behavior of target closed-source systems, to identify principal directions of unwanted knowledge representations through our SVD-based analysis, then systematically design prompts that incorporate counter-examples and directional guidance that implement our POVs concept at the prompt level. The information-theoretic principles underlying our mutual information calculations could provide crucial guidance for optimizing such unlearning prompts by quantifying the entanglement between different knowledge domains within the prompt structure itself, enabling systematic optimization of prompt templates that maximize separation between forget and retain domains within API-only constraints.