What makes unlearning hard and what to do about it

Zhang Hao
2024.6.21

What makes unlearning hard and what to do about it

Kairan Zhao
University of Warwick
Kairan.Zhao@warwick.ac.uk

Meghdad Kurmanji
University of Warwick
Meghdad.Kurmanji@warwick.ac.uk

George-Octavian Barbulescu
University of Warwick
George-Octavian.Barbulescu@warwick.ac.uk

Eleni Triantafillou*
Google DeepMind
etriantafillou@google.com

Peter Triantafillou*
University of Warwick
P.Triantafillou@warwick.ac.uk

Abstract

The article identifies and empirically examines two factors that affect the difficulty of unlearning, and then propose a Refined-Unlearning Meta-algorithm (RUM) for improving unlearning pipelines.

Evaluation

Tug-of-War(ToW)

$$\text{ToW}(\theta^u, \theta^r, \mathcal{S}, \mathcal{R}, \mathcal{D}_{test}) = (1 - \text{da}(\theta^u, \theta^r, \mathcal{S})) \cdot (1 - \text{da}(\theta^u, \theta^r, \mathcal{R})) \cdot (1 - \text{da}(\theta^u, \theta^r, \mathcal{D}_{test}))$$

$$\text{a}(\theta, \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(x,y)\in\mathcal{D}} [f(x;\theta) = y] \qquad \text{da}(\theta^u, \theta^r, \mathcal{D}) = |\text{a}(\theta^u, \mathcal{D}) - \text{a}(\theta^r, \mathcal{D})|$$

Membership Inference Attack(MIA)

$$\text{MIA Performance} = \frac{TN_{\mathcal{S}}}{|\mathcal{S}|}$$

MIA gap = |MIAretrain - MIAu|

Factor

Entanglement

$$\mathrm{ES}(\mathcal{R}, \mathcal{S}; \theta^o) = \frac{\frac{1}{|\mathcal{R}|}\sum_{i\in\mathcal{R}}(\phi_i - \mu_{\mathcal{R}})^2 + \frac{1}{|\mathcal{S}|}\sum_{j\in\mathcal{S}}(\phi_j - \mu_{\mathcal{S}})^2}{\frac{1}{2}\left((\mu_{\mathcal{R}} - \mu)^2 + (\mu_{\mathcal{S}} - \mu)^2\right)}$$

$\phi_i = g(x_i; \theta^o)$ is the embedding of example x_i

$\mu_{\mathcal{R}} = \frac{1}{|\mathcal{R}|}\sum_{i\in\mathcal{R}}\phi_i$ is the mean embedding of the retain set

$\mu_{\mathcal{S}}$ the mean embedding of the forget set

Higher ES score corresponds to higher entanglement in the embedding space.

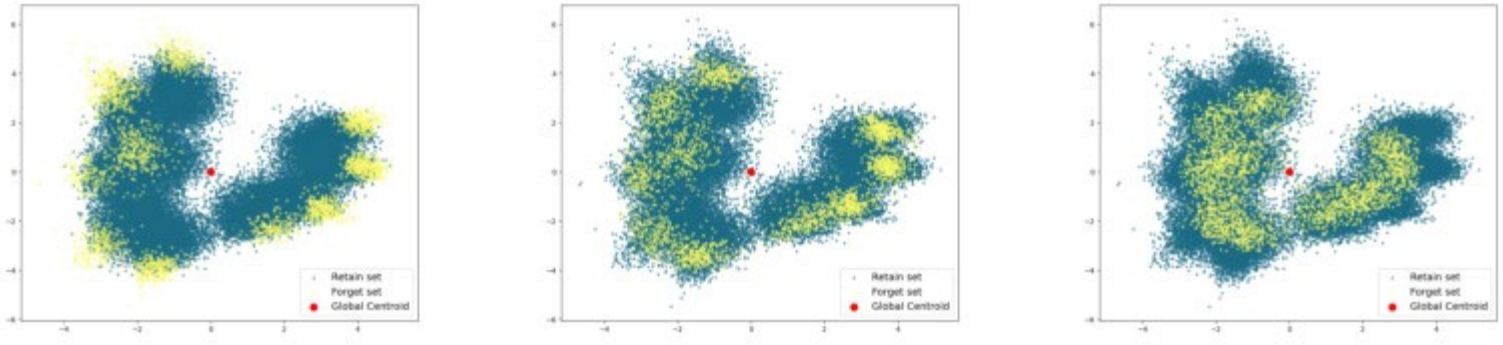
Factor

Entanglement

Procedure for creating retain / forget partitions with varying ES

	Low ES	Medium ES	High ES
ES value	309.94±98.56	1076.99±78.64	1612.210±110.82

$$d(i, \mu; \theta^o) = ||\phi_i - \mu||^2$$



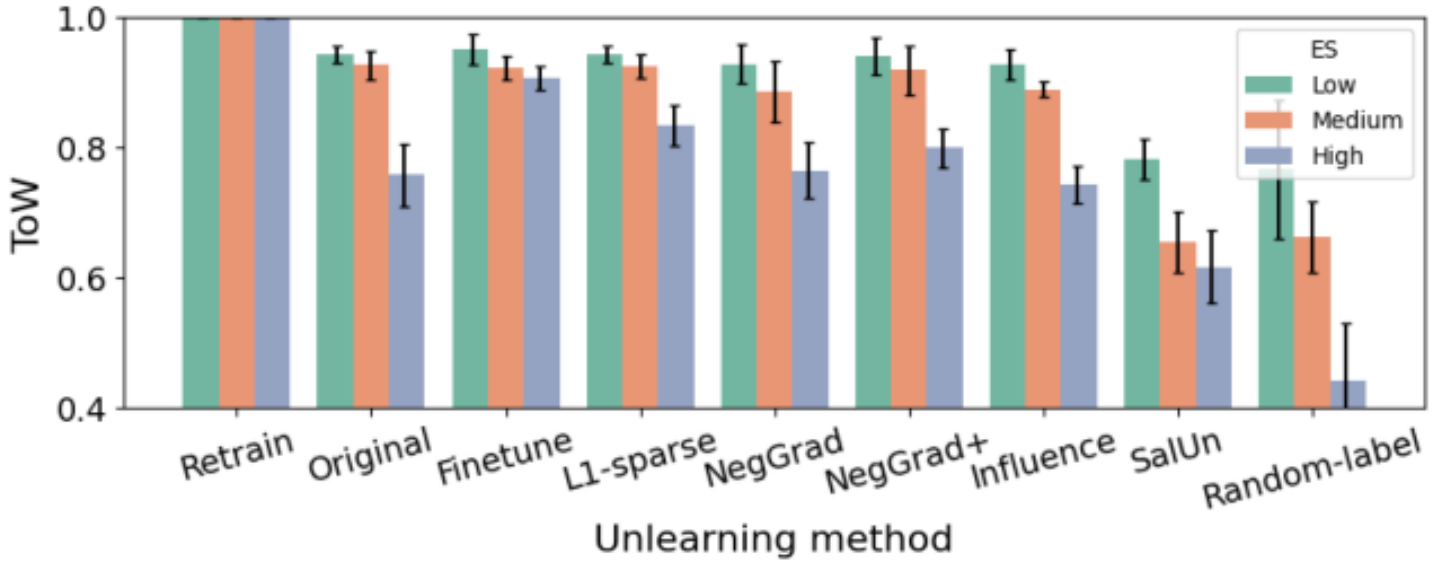
(a) low ES (b) medium ES (c) high ES

Data examples from the forget set are shown in yellow, while those from the retain set are in blue.

Factor

Entanglement

The more entangled the forget and retain sets are, the harder unlearning becomes



Factor

Memorization

$$\text{mem}(\mathcal{A}, \mathcal{D}, i) = \Pr_{f \sim \mathcal{A}(\mathcal{D})}[f(x_i) = y_i] - \Pr_{f \sim \mathcal{A}(\mathcal{D} \setminus i)}[f(x_i) = y_i]$$

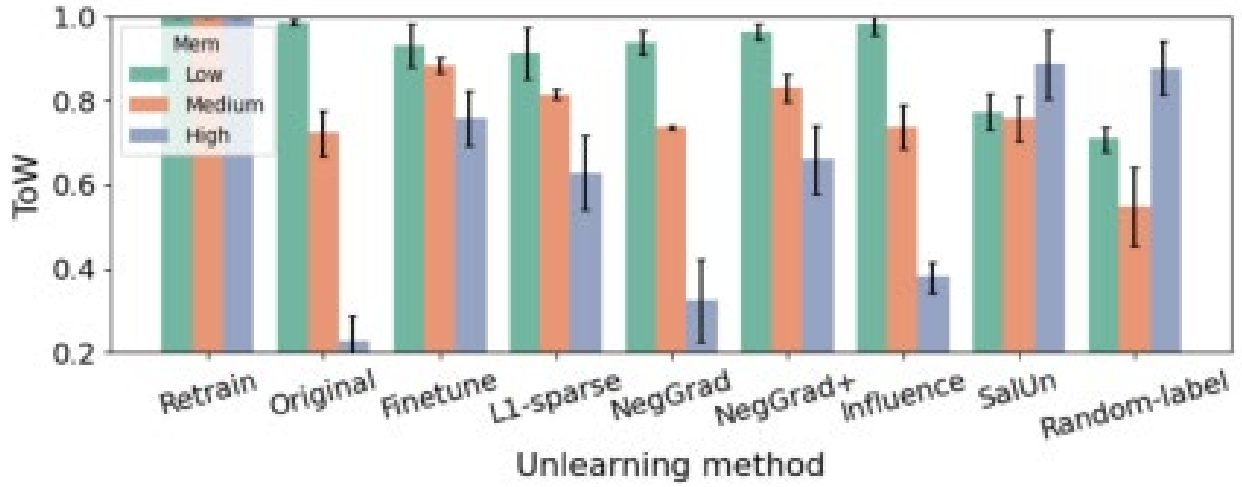
x_i and y_i are the feature and label, respectively, of example i

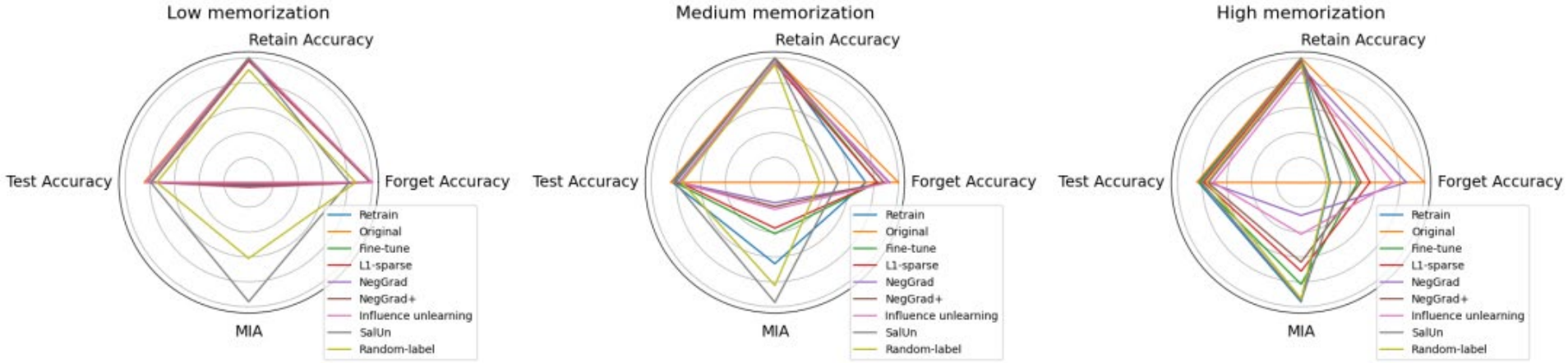
with respect to a training dataset $\mathcal{D}$ and training algorithm $\mathcal{A}$

Factor

Memorization

The more memorized the forget examples are, the harder unlearning becomes





Factor

	Low memorization	Medium memorization	High memorization
ES value	21134.127	32785.711	14736.591

ES values for forget / retain partitions across varied memorization levels

This demonstrates that embedding space entanglement and the memorization level of the forget set are distinct concepts, not merely different aspects of the same phenomenon.

Refined-Unlearning Meta-algorithm

RUM

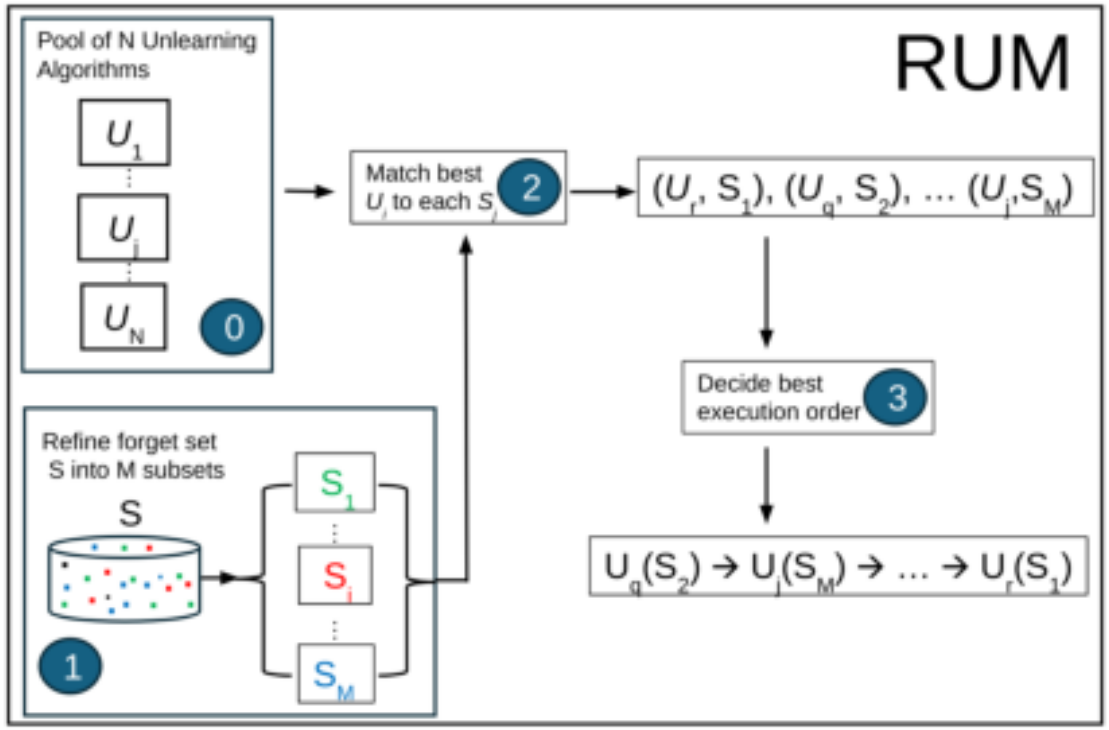
The optimal unlearning algorithm to use is dependent on the properties of the forget set

1)Refinement

Partition the forget set into homogeneous subsets w.r.t factors that affect the difficulty.

2) Meta-unlearning

Pick an unlearning algorithm for each subset and execute them in sequence .



Refined-Unlearning Meta-algorithm

RUM

	CIFAR-10		CIFAR-100	
	ToW (↑)	MIA gap (↓)	ToW (↑)	MIA gap (↓)
Retrain	1.000±0.000	0.000	1.000±0.000	0.000
Fine-tune vanilla	0.849±0.030	0.120	0.734±0.025	0.139
Fine-tune shuffle	0.712±0.040	0.098	0.589±0.036	0.345
Fine-tune RUM$^{\mathcal{F}}$	**0.937±0.052**	0.099	**0.784±0.040**	0.093
L1-sparse vanilla	0.794±0.035	0.175	0.824±0.011	0.089
L1-sparse shuffle	0.716±0.023	0.257	0.604±0.023	0.353
L1-sparse RUM$^{\mathcal{F}}$	**0.900±0.020**	0.072	**0.883±0.046**	0.033
NegGrad+ vanilla	0.802±0.028	0.230	0.861±0.069	0.159
NegGrad+ shuffle	0.632±0.022	0.520	0.613±0.054	0.417
NegGrad+ RUM$^{\mathcal{F}}$	**0.879±0.068**	0.134	**0.921±0.034**	0.059
SalUn vanilla	0.731±0.070	0.374	0.545±0.061	0.372
SalUn shuffle	0.727±0.030	0.234	0.538±0.019	0.237
SalUn RUM$^{\mathcal{F}}$	**0.887±0.069**	0.031	**0.614±0.037**	0.181
RUM	**0.965±0.014**	0.034	**0.921±0.034**	0.059

Vanilla: in one go

Shuffle: on a random partition of S into 3 equal-sized subsets

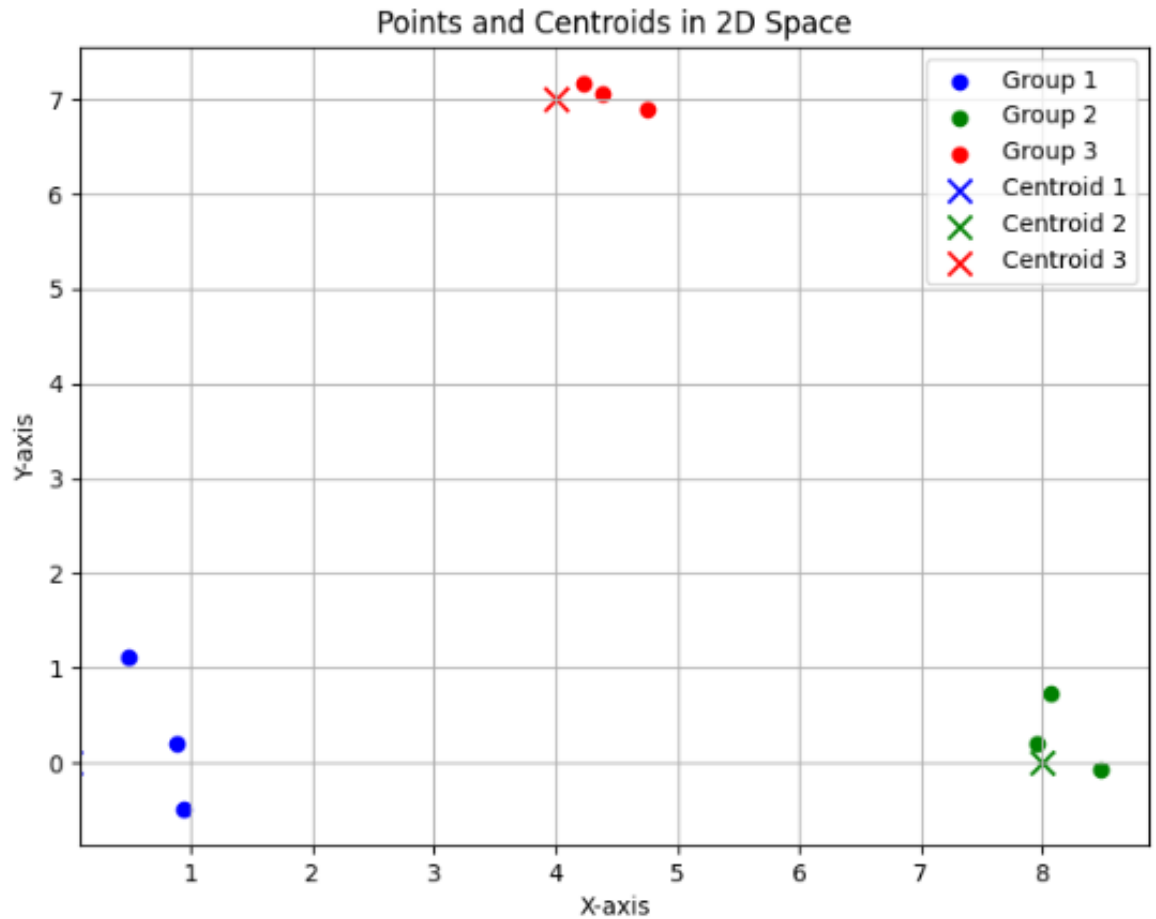
RUM$^{\mathcal{F}}$:on three equal-sized subsets obtained by F in low → med →high order

Questions

1）The background of this paper is machine unlearning. The proposed RUM framework requires fine-tuning the model for each subset during unlearning, which is too costly for large models.

2） Another issue is that the performance metrics and the division based on memorization levels proposed in this paper are based on labeled data, which are not suitable for generative tasks of large models.

3）In the entanglement part of this research, there is a constant feeling of unease about the dataset partitioning. Although the results appear effective, there is a sense that many scenarios could render them ineffective.


Points and Centroids in 2D Space
Group 1
Group 2
Group 3
Centroid 1
Centroid 2
Centroid 3
Y-axis
X-axis
0
1
2
3
4
5
6
7
8

THANKS