Unlogit: Machine Unlearning through Sensitivity-Driven Logit Control

In this supplementary supplementary document, we share experimental details including dataset and model for reproducibility purposes. We conducted 4 main experiments and 3 ablation studies. The details are below:

1. **Class unlearning with CIFAR-10 and ResNet-18**
    - **Dataset**: We used dataset split provided by SCRUB [2]. The dataset has 5K samples for each class (0-9) where 1K samples from each class constitute the Test set and the remaining 4K from each class make the training set. All the 4K samples from the training set of Class-5 created the forget dataset.
        (a) Retain set = {Class 0 to Class 4: 4000, Class 5: 0, Class 6 to Class 9: 4000, Class 5: 0}
        (b) Forget set = {Class 5: 4000}
        (c) Test set = {Class 0 to Class 9: 1000}
    - **Model**: We use ResNet-18 implementation provided by SCRUB [2] in order to have the same set up for a fair comparison.
2. **Class unlearning with TinyImageNet and ResNet-18**
    - **Dataset**: We used dataset split provided in [1]. All the samples from the training set of Class 0 to Class 4 created the forget dataset.
        (a) Retain set = {Class 0 to Class 4: 0, Class 5 to Class 199: 500}
        (b) Forget set = {Class 0 to Class 4: 500}
        (c) Test set = {Class 0 to Class 199: 25}
    - **Model**: We use ResNet-18 implementation provided in [1] in order to have the same set up for a fair comparison.
3. **Selective unlearning with CIFAR-10 and ResNet-18**
    - **Dataset**: We used dataset split provided by SCRUB [2].
        (a) Retain set = {Class 0 to Class 4: 4000, Class 5: 3900, Class 6 to Class 9: 4000}
        (b) Forget set = {Class 5: 100}
        (c) Test set = {Class 0 to Class 9: 1000}
    - **Model**: We use ResNet-18 implementation provided by SCRUB [2] in order to have the same set up for a fair comparison.
4. **Selective unlearning with CIFAR-10 and AllCNN**
    - **Dataset**: We used dataset split provided in [1].
        (a) Retain set = {Class 0: 4500, Class 1 to Class 9: 5000}
        (b) Forget set = {Class 0: 500}
        (c) Test set = {Class 0 to Class 9: 500}
    - **Model**: We use ResNet-18 implementation provided in [1] in order to have the same set up for a fair comparison.
5. **Ablation studies**: For the ablation studies, we use the same set up of class unlearning with CIFAR-10 and ResNet-18 (Main experiment 1). For probability distribution analysis, we use the same set up as main experiments 1 and 3.

In our comprehensive exploration, we employed consistent hyperparameters to ensure the reliability of our findings. We mention all the settings for all the experiments (mentioned in the main paper) in Table 1. Specifically, there are settings for (a) Reinforced forget model training, (b) Unlearn model training, (c) Final finetuning, and (d) the other general settings.

Stage	Hyperparameter	Exp1: Class Unlearning CIFAR-10 and ResNet-18	Exp2: Class Unlearning TinyImageNet and ResNet-18	Exp3: Selective Unlearning CIFAR-10 and ResNet-18	Exp4: Selective Unlearning CIFAR-10 and AllCNN
Reinforced forget model	initial model	original model (trained on retain and forget)	original model	original model	original model
	lr	0.001	0.001	0.1	0.1
	#epochs	5	5	5	5
	Optimizer	SGD	SGD	SGD	SGD
	Scheduler	ReduceLRonPlateau	ReduceLRonPlateau	ReduceLRonPlateau	ReduceLRonPlateau
Unlearning	initial model	original model	original model	original model	original model
	kl loss weight	0.5	0.5	0.5	0.5
	neg CLS loss weight	0.6	0.5	0.1	0.1
	#epochs	40	30	50	50
	lr	0.001	0.0001	0.0001	0.0001
	Optimizer	SGD	SGD	SGD	SGD
	Scheduler	ReduceLRonPlateau	ReduceLRonPlateau	Cosine	Cosine
Finetuning	#epochs	30	15	5	5
	lr	0.001	0.001	0.0001	0.0001
	weight decay	5e-5	5e-5	5e-3	5e-3
	Optimizer	SGD	SGD	AdamW	AdamW
	Scheduler	ReduceOnPlateau	ReduceOnPlateau	ReduceOnPlateau	ReduceOnPlateau
Others	momentum	0.9	0.9	0.9	0.9
	weight decay	5e-5	5e-5	5e-5	5e-5
	kl temperature	0.1	0.1	0.1	0.1

Table 1: Hyper-parameter settings for all the four main experiments.

References

[1] Tuan Hoang, Santu Rana, Sunil Gupta, and Svetha Venkatesh. Learn to unlearn for deep neural networks: Minimizing unlearning interference with gradient projection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4819–4828, 2024.

[2] Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. Towards unbounded machine unlearning. *arXiv preprint arXiv:2302.09880*, 2023.