ViT-MUL: A Baseline Study on Recent Machine Unlearning Methods Applied to Vision Transformers

Ikhyun Cho
University of Illinois at Urbana-Champaign
ihcho2@illinois.edu

Changyeon Park
Seoul National University
blackco@snu.ac.kr

Julia Hockenmaier
University of Illinois at Urbana-Champaign
juliahmr@illinois.edu

Abstract

Machine unlearning (MUL) is an arising field in machine learning that seeks to erase the learned information of specific training data points from a trained model. Despite the recent active research in MUL within computer vision, the majority of work has focused on ResNet-based models. Given that Vision Transformers (ViT) have become the predominant model architecture, a detailed study of MUL specifically tailored to ViT is essential. In this paper, we present comprehensive experiments on ViTs using recent MUL algorithms and datasets. We anticipate that our experiments, ablation studies, and findings could provide valuable insights and inspire further research in this field.

1. Introduction

Machine unlearning (MUL), aiming to eliminate knowledge of specific training data points stored in a pre-trained model, has emerged as a significant research area in deep learning. This trend is driven by the growing privacy concerns associated with large pre-trained models [5]. Furthermore, laws such as *The Right to be Forgotten* [25] and the *California Consumer Privacy Act* [21] grant users the authority to request companies to delete their privacy-related information from pre-trained models.

A straightforward solution to Machine Unlearning (MUL) is to re-train the model from scratch using the modified training data, where the data points that need to be forgotten is excluded. However, this approach is often computationally too expensive, given the large volume of training data. Therefore, the search for an efficient way to *unlearn* the pre-trained model, as opposed to retraining from scratch, has been a subject of extensive recent research.

Despite recent active research in MUL, the majority of studies have focused on ResNet-based models [1, 5, 11, 13]. However, Vision Transformer (ViT) has emerged as the dominant model architecture in various areas of computer vision. Hence, there is a crucial need for MUL studies that specifically target ViT models. In response to this need, we conduct comprehensive machine unlearning experiments on ViT models using the recently proposed MUL algorithms and datasets [5]. Specifically, we utilize two most widely-used ViT models, ViT-base and ViT-large, applying and analyzing recent machine unlearning algorithms on these architectures. We anticipate that our experiments, ablation studies, and findings could offer valuable insights and motivate further research in this field. Code is available at https://github.com/ihcho2/ViTMUL.

2. Related Work

2.1. Vision Transformer (ViT)

Transformers [24], initially prevalent in natural language processing [3, 4, 6, 15, 19, 22], have now emerged as a dominant model architecture in computer vision as well. Specifically, ViT [7] forms the basis of current state-of-the-art techniques in various tasks (e.g., image classification, image segmentation, image captioning e.t.c.) [17, 20, 26]. Unlike traditional approaches, ViT exclusively employs a transformer architecture. This process involves dividing images into fixed-size patches (tokens), adding learnable position embeddings to each token, which are then processed through multiple layers of self-attention-based transformers. This method enables the learning of relationships between the segmented image parts in a parallelizable way, ultimately leading to an effective image understanding. ViT models, pre-trained on millions of image-context pairs, have now become the de facto models used in a variety of fields in computer vision [17, 20]. Therefore, it is imperative to test machine unlearning methods on ViT-based models.

2.2. Machine Unlearning

Background The concept of Machine Unlearning was initially introduced by [2], who utilized statistical query

learning in its development. Subsequently, a significant body of research, as seen in studies [9, 10], has concentrated on devising machine unlearning methodologies specifically for deep neural networks. In this paper, we undertake a comparative analysis of various machine unlearning approaches, with the aim of establishing a foundational baseline for machine unlearning in ViT systems.

Formal definition and metrics Given a pre-trained model θ_0 initially trained on D_T, suppose we want to make θ_0 forget a set of data D_F that is part of the training data (i.e., $D_F \subset D_T$). Here, D_F is referred to as the forget set, and the complement $D_R (= D_T \cap D_F^c)$ is termed the retain set. The goal of machine unlearning is to create an unlearning algorithm U in such a way that, upon application, the unlearned model, $\theta^* = U(\theta_0, D_F, D_R)$, effectively forgets D_F. Machine unlearning algorithms are typically assessed using two key metrics: *utility* and *forgetting performance*.

Utility is evaluated by assessing the model's accuracy on a distinct test set, D_{Test}. A valuable unlearning algorithm should ensure that the overall performance of the model does not degrade significantly. Hence, we expect θ^* to have a similar (or ideally an improved) test accuracy compared to θ_0.

To assess the forgetting performance, an additional distinct test set D_{Unseen} is used. At a high level, the objective is to ensure that the unlearned model's behavior on the forget set D_F closely resembles its behavior on D_{Unseen}. To achieve this, a classifier, such as a regression model, is typically trained to distinguish between the outputs, $\theta^*(D_F)$ and $\theta^*(D_{Unseen})$. That is, we collect the losses from $\theta^*(D_F)$ and $\theta^*(D_{Unseen})$, and then train a regression classifier to distinguish between them. If the classifier's accuracy is close to 50% (i.e., the classifier is unable to distinguish them), it indicates a high similarity between the outputs $\theta^*(D_F)$ and $\theta^*(D_{Unseen})$, suggesting that the model has effective forgot D_F (more details can be found in [5]).

Intuitively, if a model efficiently unlearns specific data, the general performance is likely to decrease, especially when dealing with a large amount of forget data. Consistent with this intuition, a general trade-off exists between the model's utility and its forgetting performance [5, 11]. The primary goal of MUL is to derive a decent trade-off between them.

2.3. Instance-based Machine Unlearning

Most previous MUL studies have used conventional computer vision datasets such as MNIST [14], CIFAR-10 [12], and SVHN [18]. Prior research focused on unlearning specific class(es) (e.g., unlearning numbers of 9 in MNIST). However, recent observations indicate that this setting does not align well with real-world applications [5]. In practice, the need often arises to forget specific instances (individuals) that may have different labels, rather than an entire class. Acknowledging this, Choi and Na [5] introduced two new machine unlearning benchmark datasets: MUFAC and MUCAC. In this paper, our focus is on instance-based machine unlearning, as it is more aligned with real-world scenarios.

MUFAC (Machine Unlearning for Facial Age Classifier) Choi and Na [5] introduced a multi-class age classification dataset, MUFAC, comprising Asian facial images with annotated labels. The labels categorize individuals into eight age groups. Unlike previous machine unlearning tasks that aim to forget specific class(es), MUFAC focuses on unlearning a group of individuals with various class labels. Similar to previous MUL tasks, the objective is to achieve a balanced trade-off between test accuracy and forgetting quality.

MUCAC (Machine Unlearning for Celebrity Attribute Classifier) Choi and Na [5] introduced another MUL dataset derived from CelebA [16], called MUCAC, which involves a multi-label facial classification task. The labels contain three attributes: gender (male/female), age (old/young), and expression (smiling/unsmiling). Statistical details of MUFAC and MUCAC can be found in Section 3.1.

2.4. Baselines

We provide an overview of well-known machine unlearning algorithms widely used in recent days below.

Fine-Tuning Refining the original model through fine-tuning, utilizing solely the retain set, can be an effective approach. During this learning process, employing a marginally higher learning rate can enhance generalization. This adjustment may lead to more effective forgetting of the data intended to be unlearned [10].

Catastrophically Forgetting-k The concept of Catastrophically Forgetting in the last k layers (CF-k), as proposed by [9], involves fine-tuning only the last k layers while keeping the rest unchanged. This method leverages the phenomenon of catastrophic forgetting in machine learning models, as documented in [8]. Catastrophic forgetting occurs when a model is repeatedly updated without retraining on previously learned data, leading to a gradual loss of information related to that data. By focusing the learning process on the last k layers, this approach requires fewer training epochs and is able to maintain the utility of the model more effectively.

Advanced Negative Gradient Introduced by [5], Advanced Negative Gradient (AdvNegGrad) is an enhanced version of Negative Gradient (NegGrad) [10]. While NegGrad applies gradient ascent using the forget set to increase loss, leading to data oblivion, AdvNegGrad integrates the joint loss of fine-tuning with NegGrad's approach within the same training batches.

Unlearning by Selective Impair and Repair The Un-

learning by Selective Impair and Repair (UNSIR) method, proposed by [23], initially designed to forget specific classes in a model, introduces disruptive noise to negatively impact the model's weights during the learning phase. After this corruption, the model undergoes fine-tuning with a retain set to rectify these changes. However, unlike its original purpose, in this paper [5], UNSIR is adapted to forget specific individual data points. This is achieved by learning a synthesized noise that maximizes the difference from the data to be forgotten, followed by fine-tuning to correct the altered weights.

SCalable Remembering and Unlearning unBound In this context, a technique is used where the knowledge of the forget set is removed by employing a stochastic initialization model, which serves as a student model. This method differs from joint training as it uses a "bad teacher" concept to eliminate the impact of the forget set. However, this approach of maximizing the distance between the student and teacher models for the forget set can adversely affect the performance on the retain set. To address this issue, [13] introduced SCalable Remembering and Unlearning unBound (SCRUB). This method aims to maintain the student model's closeness to the teacher on the retain set while distancing it on the forget set.

Attack-and-Reset Presented by [11], Attack-and-Reset (ARU) method identifies and re-initializes parameters in a model that are prone to overfitting on the forget set. It determines these parameters by measuring the gradient differences between the original forget images and noise. If this gradient discrepancy is small, it indicates that these parameters are responsible to the model's overfitting. By re-initializing these specific parameters and applying fine-tuning afterwards, ARU leads to the model forgetting the data in the forget set.

In this paper, we apply the aforementioned baseline algorithms to Vision Transformer (ViT)-based models and assess their performance in an instance-based machine unlearning setting using the MUFAC and MUCAC datasets.

3. Experiments

3.1. Datasets

As explained in Section 2.3, we focus on unlearning instances rather than unlearning a class(es), as it better aligns with real-world applications. Hence, we use the recently introduced datasets, MUFAC and MUCAC, provided by Choi and Na [5]. Statistics of these datasets are summarized in Table 1. All images in MUFAC and MUCAC have a resolution of 128×128, focusing on the facial region.

	MUFAC	MUCAC
Train dataset	10,025	25,846
Test dataset	1,539	2,053
Forget dataset	1,500	10,135
Retain dataset	8,525	15,711
Unseen dataset	1,504	2,001

Table 1. Overall statistics of MUFAC and MUCAC.

3.2. Experimental Settings

We use two of the most widely used ViT models, ViT-Base and ViT-Large. For faster convergence and better overall performance, we use the pre-trained models (ViT-B-16 and ViT-L-14) provided by Radford et al. [20]. We follow the default hyper-parameter settings from the official repository [20] (i.e., optimizer of adamW, learning rate of 1e-5, batch size of 64, patch size of 16x16 for ViT-B-16 and 14x14 for ViT-L-14). We use 30 epochs for unlearning, recording the best outcome based on the NoMUS score. For accurate evaluations, we use 5 random seeds equally for all algorithms and report the average along with standard deviations.

3.3. Overall Results

Table 2 presents the results of baseline unlearning techniques applied to ResNet18 and Vision Transformers (ViT-B-16 and ViT-L-14). Based on the results, we observe several findings: (1) Vision Transformers (ViT) show better general performance compared to ResNet18, with ViT-L-14 outperforming ViT-B-16 (e.g., pre-trained models on MUFAC show accuracies of 59.52% for ResNet18, 66.54% for ViT-B-16, and 71.35% for ViT-L-14). This is consistent with the general trend where ViT models are generally superior to ResNets, and the larger ViT-L model outperforms the smaller ViT-B model; (2) All the baseline unlearning methods are effective on ViT models, as evidenced by the increase in the NoMUS score from the pre-trained model; (3) algorithms that are relatively more effective on ResNet18 (e.g., ARU, AdvNegGrad, SCRUB) also show greater effectiveness on ViTs, indicating a general trend between ResNet and ViT.

3.4. Specific Results

We provide a concise analysis of each of the unlearning algorithms.

Re-training from scratch As explained in Section 2.4, re-training from scratch (denoted by "Re-train" in Table 2) serves as the foundational baseline method for machine unlearning. In all four scenarios ({ViT-B, ViT-L} x {MUFAC, MUCAC}), the re-training method shows improved forgetting performance as expected. However, within our setting,

	MUFAC			MUCAC		
Model	Utility (%, ↑)	Forget (%, ↓)	NoMUS (%, ↑)	Utility (%, ↑)	Forget (%, ↓)	NoMUS (%, ↑)
1. ResNet18						
Pre-trained	59.52	21.36	58.40	88.52	4.19	90.07
Unlearning						
• Re-train	47.34 (±0.84)	3.09 (±1.06)	70.58 (±0.92)	87.62 (±3.38)	3.03 (±1.52)	90.77 (±1.68)
• Finetune	59.57 (±0.43)	19.89 (±0.16)	59.90 (±0.31)	91.05 (±0.78)	3.17 (±0.61)	92.35 (±0.66)
• CF-k	59.42 (±0.55)	20.11 (±0.28)	59.60 (±0.39)	91.96 (±0.14)	4.29 (±0.55)	91.69 (±0.58)
• AdvNegGrad	49.37 (±0.63)	0.56 (±0.68)	74.13 (±0.78)	88.51 (±2.58)	3.32 (±0.61)	90.93 (±1.24)
• UNSIR	59.07 (±0.40)	20.27 (±0.27)	59.27 (±0.36)	91.98 (±0.40)	3.61 (±0.45)	92.38 (±0.40)
, • SCRUB	52.45 (±1.05)	0.99 (±0.73)	75.23 (±0.71)	90.09 (±1.61)	2.62 (±1.11)	92.43 (±0.49)
, • ARU	59.25 (±1.31)	0.61 (±0.42)	79.01 (±0.49)	90.33 (±0.74)	2.00 (±0.62)	93.17 (±0.59)
2. ViT-B-16						
Pre-trained	66.54	12.56	70.71	95.74	8.95	88.92
Unlearning						
• Re-train	62.70 (±2.28)	5.94 (±1.20)	75.40 (±1.42)	95.07 (±0.26)	2.10 (±0.41)	95.43 (±0.40)
• Finetune	64.78 (±0.79)	2.03 (±1.05)	80.36 (±0.81)	94.50 (±0.71)	3.30 (±0.39)	93.95 (±0.18)
• CF-k	64.98 (±0.89)	2.37 (±1.27)	80.12 (±0.99)	94.91 (±0.18)	3.78 (±0.30)	93.67 (±0.26)
• AdvNegGrad	63.10 (±1.26)	1.04 (±0.97)	80.51 (±0.85)	94.08 (±0.13)	0.42 (±0.43)	96.62 (±0.44)
• UNSIR	64.93 (±0.90)	2.26 (±0.76)	80.20 (±0.74)	94.72 (±0.31)	3.73 (±0.58)	93.62 (±0.58)
, • SCRUB	65.93 (±0.84)	1.45 (±0.76)	81.52 (±0.42)	93.74 (±2.21)	4.03 (±1.38)	92.84 (±0.34)
• ARU	62.44 (±0.62)	0.96 (±0.94)	80.26 (±0.92)	94.63 (±0.14)	2.96 (±0.73)	94.36 (±0.76)
3. ViT-L-14						
Pre-trained	71.35	18.9	66.77	95.63	6.58	91.23
Unlearning						
• Re-train	67.23 (±1.49)	5.05 (±0.64)	78.56 (±0.95)	94.88 (±0.14)	1.25 (±0.27)	96.19 (±0.24)
• Finetune	67.75 (±1.40)	8.34 (±1.14)	75.53 (±0.56)	94.97 (±0.21)	4.22 (±0.27)	93.27 (±0.24)
• CF-k	68.98 (±1.15)	10.13 (±1.07)	74.36 (±1.11)	94.85 (±0.10)	4.41 (±0.27)	93.01 (±0.30)
• AdvNegGrad	66.33 (±1.84)	2.76 (±1.24)	80.40 (±0.98)	94.28 (±0.22)	0.41 (±0.31)	96.72 (±0.27)
• UNSIR	66.81 (±1.86)	5.42 (±1.19)	77.98 (±0.62)	94.52 (±0.38)	2.42 (±0.29)	94.84 (±0.34)
, • SCRUB	67.28 (±3.38)	2.92 (±2.06)	80.72 (±1.14)	94.42 (±0.89)	3.51 (±0.99)	93.70 (±0.60)
• ARU	65.07 (±1.38)	0.53 (±0.46)	82.01 (±0.59)	94.98 (±0.14)	2.87 (±0.39)	94.62 (±0.37)

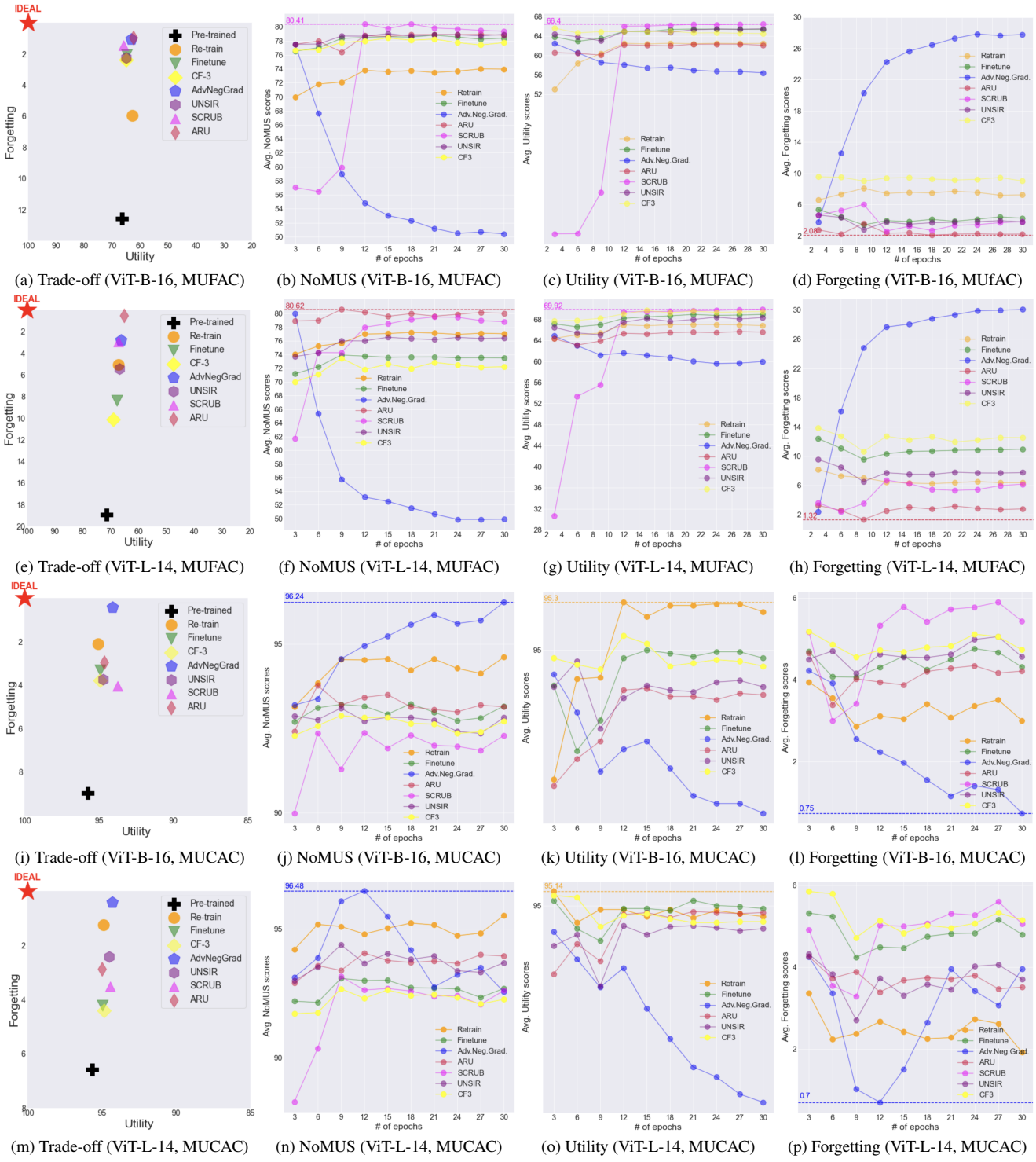
Table 2. Overall results of unlearning experiments on MUFAC and MUCAC benchmarks using ViT-based models. ↑ / ↓ indicates the metrics being larger/smaller the better, respectively.

where the forget dataset is relatively large, re-training from scratch generally yields a lower utility score, consequently resulting in a relatively lower overall NoMUS score. Thus, in situations where the forget data is extensive, opting for alternative unlearning algorithms that leverage the pre-trained θ_0 instead of re-training from scratch, could be a wiser choice.

Finetuning on the retain set Fine-tuning on the retain set (indicated as "Finetune" in Table 2) proves to be effective in all four settings, particularly in forgetting performance. However, it is relatively less powerful in forgetting compared to re-training. This outcome aligns with expectations since, even though it undergoes fine-tuning on the retain set, the traces of the forget set do not fully vanish.

CF-k Fine-tuning only the final k layers of the original model on the retain dataset (indicated as "CF-k" in Table 2) also demonstrates effectiveness in all four settings. An ablation study, as presented in Table 3, verifies that larger values of k are more effective. It's worth noting that, in all four cases, CF-k does not outperform simple fine-tuning. We observe that freezing the lower layers is not effective in achieving a satisfactory forgetting performance.

Advanced Negative Gradient Introduced by [5], Advanced Negative Gradient (AdvNegGrad) leverages gradi-



(a) Trade-off (ViT-B-16, MUFAC) (b) NoMUS (ViT-B-16, MUFAC) (c) Utility (ViT-B-16, MUFAC) (d) Forgeting (ViT-B-16, MUfAC)

(e) Trade-off (ViT-L-14, MUFAC) (f) NoMUS (ViT-L-14, MUFAC) (g) Utility (ViT-L-14, MUFAC) (h) Forgetting (ViT-L-14, MUFAC)

(i) Trade-off (ViT-B-16, MUCAC) (j) NoMUS (ViT-B-16, MUCAC) (k) Utility (ViT-B-16, MUCAC) (l) Forgetting (ViT-B-16, MUCAC)

(m) Trade-off (ViT-L-14, MUCAC) (n) NoMUS (ViT-L-14, MUCAC) (o) Utility (ViT-L-14, MUCAC) (p) Forgetting (ViT-L-14, MUCAC)

Figure 1. Visualization results of each unlearning algorithms' performance over the epoch. Each mark in the plot represents the average of 5 seed runs. The first column provides a summary of the trade-off between utility and forgetting for each algorithm.

ent ascent on the forget set to mitigate overfitting. Consistent with its underlying motivation, AdvNegGrad shows notable forgetting performance on both datasets, for both ViT-B-16 and ViT-L-14 models. These outcomes underscore the effectiveness of integrating gradient ascent for effective forgetting. One limitation is that the model's performance is

highly unstable due to the gradient ascent term becoming unboundedly large as unlearning progresses. Balancing the two loss terms and inducing a stable unlearning would be an interesting future work.

UNSIR Leveraging noise to counteract the overfitting of the model to the forget set (denoted as "UNSIR" in Table 2) proves to be effective in all four cases as well. However, when compared to other strong baselines like AdvNegGrad and SCRUB, the overall performance is slightly inferior.

SCRUB Similar to AdvNegGrad, the SCRUB method also utilizes gradient ascent on the forget set. SCRUB introduces a coefficient as its hyper-parameter, which balances the two losses (gradient descent on D_R and gradient ascent on D_F). We perform a grid search on this coefficient hyperparameter, and the outcomes are summarized in Table 3. The results highlight the significant influence of this hyperparameter on performance, necessitating the need for hyperparameter search when applying SCRUB to new tasks or models.

ARU The Attack-and-Reset [11] approach is a re-initialization-based method designed to effectively identify and reset parameters that are responsible for overfitting to the forget set. ARU demonstrates state-of-the-art performance on both MUFAC and MUCAC when applied to ResNet18. Similarly, the method is also effective on ViTs. ARU has a unique hyperparameter, the pruning ratio (ranging from 0% to 100%), where the final performance heavily depends on. We conducted a fundamental ablation study which involves exploring different resetting ratios (i.e., 10%, 30%, 50%, 70%, 90%). We used ViT-B-16 as the representative model.

Table 3 summarizes the results, revealing two main findings: (1) ARU's performance is highly dependent on the pruning ratio when applied to Vision Transformer (ViT) models; (2) pruning more than 10% of the ViT parameters lead to considerable degradation in performance, while the optimal pruning rate being 50% for ResNet18. This disparity suggests that ViT-based models might employ parameters more efficiently, leading to lower redundancy.

3.5. Visualization

We present visualization results in Figure 1 to provide insights into how each algorithm facilitates the unlearning process. We observe several interesting findings: (1) Most algorithms converge to their specific performance after approximately 15 epochs, indicating that running unlearning for 30 epochs could be redundant; (2) AdvNegGrad, which is overall the most effective unlearning algorithm, generally takes more time to reach the peak compared to other methods; (3) Methods that utilize gradient ascent (i.e., SCRUB and AdvNegGrad) become relatively unstable, which we speculate is due to the gradient ascent term easily becoming the dominant loss term.

	MUFAC		
Model (ViT-B-16)	Utility (%, ↑)	Forget (%, ↓)	NoMUS (%, ↑)
CF-k			
• k: 3	65.21 (±0.56)	7.81 (±0.64)	74.79 (±0.72)
• k: 6	64.91 (±1.16)	4.36 (±1.07)	78.09 (±0.86)
• k: 9	64.98 (±0.89)	2.37 (±1.27)	80.12 (±0.99)
ARU			
• pruning ratio: 10%	62.44 (±0.62)	0.96 (±0.94)	80.26 (±0.92)
• pruning ratio: 30%	53.03 (±1.56)	5.95 (±0.81)	70.57 (±0.98)
• pruning ratio: 50%	37.57 (±1.33)	3.76 (±0.93)	65.02 (±0.66)
• pruning ratio: 70%	31.06 (±0.95)	3.91 (±0.60)	61.62 (±0.43)
• pruning ratio: 90%	30.10 (±1.05)	5.07 (±0.81)	59.99 (±1.08)
SCRUB			
• coefficient: 1.0	33.76 (±5.89)	4.73 (±1.34)	62.15 (±2.11)
• coefficient: 0.1	65.93 (±0.84)	1.45 (±0.76)	81.52 (±0.42)
• coefficient: 0.01	63.22 (±2.73)	1.71 (±1.88)	79.90 (±1.12)

Table 3. Ablation studies on the effect of different hyperparameter values on ARU and CF-k.

4. Conclusion

This paper presents a baseline study on recent machine unlearning approaches applied to Vision Transformer (ViT) models using the recently proposed machine unlearning datasets [5]. As ViT models are becoming more and more dominant these days, a comprehensive baseline study of recent unlearning methods on ViT models becomes crucial for future research in this field. We hope our work could provide insights and contribute to an active future research in the field.

References

[1] Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE, 2021. 1

[2] Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE, 2015. 1

[3] Ikhyun Cho and U Kang. Pea-kd: Parameter-efficient and accurate knowledge distillation on bert. *Plos one*, 17(2): e0263592, 2022. 1

[4] Ikhyun Cho, Yoonhwa Jung, and Julia Hockenmaier. Sirabsc: Incorporating syntax into roberta-based sentiment analysis models with a special aggregator token. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8535–8550, 2023. 1

[5] Dasol Choi and Dongbin Na. Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems. *arXiv preprint arXiv:2311.02240*, 2023. 1, 2, 3, 4, 6

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arxiv 2020. *arXiv preprint arXiv:2010.11929*, 2010. 1

[8] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999. 2

[9] Shashwat Goel, Ameya Prabhu, Amartya Sanyal, Ser-Nam Lim, Philip Torr, and Ponnurangam Kumaraguru. Towards adversarial evaluations for inexact machine unlearning. *arXiv preprint arXiv:2201.06640*, 2022. 2

[10] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312, 2020. 2

[11] Yoonhwa Jung, Ikhyun Cho, Shun-Hsiang Hsu, and Julia Hockenmaier. Attack and reset for unlearning: Exploiting adversarial noise toward machine unlearning through parameter re-initialization. *arXiv preprint arXiv:2401.08998*, 2024. 1, 2, 3, 6

[12] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 2

[13] Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. Towards unbounded machine unlearning. *arXiv preprint arXiv:2302.09880*, 2023. 1, 3

[14] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1

[16] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018. 2

[17] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 1

[18] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 2

[19] Tairen Piao, Ikhyun Cho, and U Kang. Sensimix: Sensitivity-aware 8-bit index & 1-bit value mixed precision quantization for bert compression. *PloS one*, 17(4): e0265621, 2022. 1

[20] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 3

[21] State of California Department of Justice. California consumer privacy act of 2018, 2018. 1

[22] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*, 2019. 1

[23] Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 3

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1

[25] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017. 1

[26] Xiaowei Yu, Yao Xue, Lu Zhang, Li Wang, Tianming Liu, and Dajiang Zhu. Noisynn: Exploring the influence of information entropy change in learning systems, 2023. 1