LABEL-AGNOSTIC FORGETTING: A SUPERVISION-FREE UNLEARNING IN DEEP MODELS

Shaofei Shen[1] **Chenhao Zhang**[1] **Yawen Zhao**[1] **Alina Bialkowski**[1]
Weitong Chen[2] **Miao Xu**[1]*
[1]University of Queensland [2]University of Adelaide
{shaofei.shen,chenhao.zhang,yawen.zhao,alina.bialkowski}@uq.edu.au
t.chen@adelaide.edu.au, miao.xu@uq.edu.au

ABSTRACT

Machine unlearning aims to remove information derived from forgotten data while preserving that of the remaining dataset in a well-trained model. With the increasing emphasis on data privacy, several approaches to machine unlearning have emerged. However, these methods typically rely on complete supervision throughout the unlearning process. Unfortunately, obtaining such supervision, whether for the forgetting or remaining data, can be impractical due to the substantial cost associated with annotating real-world datasets. This challenge prompts us to propose a supervision-free unlearning approach that operates without the need for labels during the unlearning process. Specifically, we introduce a variational approach to approximate the distribution of representations for the remaining data. Leveraging this approximation, we adapt the original model to eliminate information from the forgotten data at the representation level. To further address the issue of lacking supervision information, which hinders alignment with ground truth, we introduce a contrastive loss to facilitate the matching of representations between the remaining data and those of the original model, thus preserving predictive performance. Experimental results across various unlearning tasks demonstrate the effectiveness of our proposed method, Label-Agnostic Forgetting (LAF) without using any labels, which achieves comparable performance to state-of-the-art methods that rely on full supervision information. Furthermore, our approach excels in semi-supervised scenarios, leveraging limited supervision information to outperform fully supervised baselines. This work not only showcases the viability of supervision-free unlearning in deep models but also opens up a new possibility for future research in unlearning at the representation level.

1 INTRODUCTION

Currently, machine unlearning has attracted increasing attention due to rising concerns about data privacy issues (Xu et al., 2024; Nguyen et al., 2022; Bourtoule et al., 2021). To protect the privacy and interest of the owner of sensitive data, legislators in many regions have introduced laws like GDPR (Voigt & Von dem Bussche, 2017), and CCPA (de la Torre, 2018), which demand the deletion of sensitive information from the well-trained models.

The objective of machine unlearning is to remove information associated with *forgetting data* from the original model while preserving the knowledge contained in the *remaining data* (Bourtoule et al., 2021). A direct and intuitive strategy to achieve this is to retrain a new model from scratch utilizing only the remaining dataset. However, this method can be both time-consuming and computationally demanding (Zhang et al., 2022; Di et al., 2022). Without doing retraining, existing works on machine unlearning can be divided into two types. The first type is *exact unlearning* (Bourtoule et al., 2021; Kim & Woo, 2022). This approach necessitates that the unlearned model attains exactly the same performance as the retrained model, with respect to both model parameters and prediction accuracy (Bourtoule et al., 2021). In deep learning models, the exact unlearning is usually achieved through a distributed retraining strategy (Bourtoule et al., 2021). The second type is termed as *approximate*

*Corresponding author

unlearning, which requires the unlearned model to get similar prediction performances to the retrained model on not only the remaining data but also the forgetting data (Thudi et al., 2022; Chundawat et al., 2023; Kurmanji et al., 2023). Current strategies for approximate unlearning encompass methods such as training teacher models to facilitate the removal of forgetting data (Chundawat et al., 2023; Kurmanji et al., 2023), or retracing the alterations of parameters occurring in the training of forgetting data to reverse its effect of training (Thudi et al., 2022). Although both methods have achieved notable performance, it is worth noting that both types of work rely on the annotated remaining and forgetting data to guide the removal of undesired information and preservation of other necessary information. This line of works can be regarded as *supervised unlearning*.

While both types of work have demonstrated commendable performance, it is imperative to acknowledge the prevalent reality: in the real world, a significant portion of data remains unannotated, leading to a substantial number of machine learning models being trained on weakly labelled data (Nodet et al., 2020). This situation is exemplified in semi-supervised learning, which capitalizes on a vast pool of unlabelled data alongside a smaller set of annotated data for training purposes (Yang et al., 2023). Moreover, in the pursuit of privacy protection, even when training data is fully labelled, these labels may not be accessible during the unlearning phase. Previous unlearning works have necessitated the use of label information as optimization targets, either for purging information related to forgotten data or for retaining knowledge about the remaining data (Thudi et al., 2022; Chundawat et al., 2023; Chen et al., 2023; Kurmanji et al., 2023). Consequently, studies focused on supervised unlearning are limited in their ability to execute the unlearning task or preserve prediction performance in the absence of sufficient supervision information. This underscores the critical need for an unlearning algorithm that operates without relying on supervision information during the unlearning process, which we term as *label-agnostic unlearning*.

Therefore, we propose a framework named Label-Agnostic Forgetting (LAF)[1] for the label-agnostic unlearning, which can accomplish the unlearning task without the supervision information. Specifically, we alleviate the dependency of the unlearning process on supervision information by adjusting the representation distributions of the representation extractor, rather than changing the classifiers. We utilize a variational inference approach (Kingma & Welling, 2014) to estimate the representation distribution of the remaining data and then design an extractor unlearning loss to adjust the representation extractor, aiming to eliminate the information associated with the forgetting data at the representation level. To retain the prediction performances of the model after shifting representations, we devise a contrastive loss to align the remaining data representations of the adjusted model with those of the original model. Furthermore, if the supervision information is available, we can further adjust the classifier to fit the output distributions with the ground truths.

The contributions of this paper can be summarised as follows:

- Addressing the research gap in label-agnostic unlearning, we introduce and propose a framework named LAF, which is capable of accomplishing unlearning tasks and retaining high predictive performance post-learning, all without the need for supervision information. The proposed LAF can work effectively for mainstream unlearning problems.
- We incorporate the variational inference and contrastive learning approaches and propose two novel loss functions for extractor unlearning and representation alignment.
- Through empirical evaluations across multiple datasets and models, we demonstrate that LAF is comparable with full supervised unlearning methods. In addition, when limited supervision information is available, the LAF can outperform other state-of-the-art works.

2 PRELIMINARY

Let D denote the training data which can be either fully supervised or semi-supervised, and g_D represents a deep model trained on D, which maps an instance $x \in \mathcal{X}$ to a label $y \in \mathcal{Y}$. The unlearning on the deep model g_D aims to remove the knowledge related to forgetting data $D_f \subset D$, and preserve the knowledge learned from the remaining data $D_r = D - D_f$. Assuming that the training data D, remaining data D_r and forgetting data D_f are sampled from the distributions $\mathcal{P}$, $\mathcal{P}_r$ and $\mathcal{P}_f$ respectively, the machine unlearning algorithm U should yield a deep model $g_U = U(g_D, D_r, D_f)$ which approximates the performance of g_{D_r} trained on D_r only. That is, $g_U(x) = g_{D_r}(x)$ no matter x is sampled from $\mathcal{P}_r$ or $\mathcal{P}_f$. From an intuitive sense, $g_U(x)$ should be similar to $g_D(x)$ for $x \sim \mathcal{P}_r$,

[1] https://github.com/ShaofeiShen768/LAF

but (significantly) different for $x \sim \mathcal{P}_f$, such that the forgetting effect can be achieved without impacting performance on the remaining data.

As a deep model, g can be viewed as the concatenation of two main components: a representation extractor g_D^e and a downstream classifier g_D^c. In the case of label-agnostic unlearning, wherein labels may not be readily accessible or reliable during the unlearning phase, a potential approach to address this challenge is to adjust the representation extractor g_D^e to eliminate the memory of forgotten data.

There are three main challenges when adjusting g_D^e. The first involves estimating the knowledge associated with forgetting data within the representation extractor g_D^e. Secondly, the process of removing the knowledge of the forgetting data from g_D^e lacks a well-defined optimization objective. Traditional objectives used in supervised unlearning scenarios may not apply, especially in cases where label information is either unavailable or unreliable in label-agnostic unlearning situations. Thirdly, modifying g_D^e can also impact the representations of the remaining data, potentially causing a misalignment with the classifier g_D^c and consequently leading to a decrease in predictive performance.

3 METHODOLOGY

In this section, we tackle the three aforementioned challenges by introducing the Label Agnostic Forgetting (LAF) method, which comprises two updates: the extractor unlearning and the representation alignment. Importantly, both of these updates operate without the need for supervision information. During the extractor unlearning stage, we estimate the distribution of representations for both the forgetting data and the remaining data, leveraging the original model's knowledge acquired from these distinct data groups. Subsequently, we introduce two objectives to facilitate unlearning, with another proposed extractor unlearning loss. Moving on to the representation alignment stage, we recognize that alterations in representation may impact the alignment between these representations and the classifiers. To address this, we propose a contrastive loss that aligns the representations post-unlearning with those pre-unlearning, preserving predictive performance in light of the absence of label information. Furthermore, we consider scenarios where limited supervision information is available. In such cases, we incorporate an additional supervised repair step to further enhance the unlearning performance.

The subsequent subsections will delve into the specifics of extractor unlearning, and representation alignment, and provide an overview of the complete LAF algorithm.

3.1 EXTRACTOR UNLEARNING

In the extractor unlearning stage, we first discuss the relationship between the data's distribution and post-unlearning extractor $g_U^e(\cdot)$. Assuming that for $x \sim \mathcal{P}_r$, $g_U^e(x)$ follows a distribution $Q(D_r)$ and for $x \sim \mathcal{P}_f$, $g_U^e(x)$ follows a distribution $Q(D_f)$, then one possible way to learn the optimal θ^* for g_U^e parameterized on θ can be two-objective, that is,

$$\min_\theta \Delta(Q(D_r), \mathcal{P}_r), \text{ where } x \sim \mathcal{P}_r, g_U^e(x) \sim Q(D_r), \text{ and simultaneously} \tag{1}$$

$$\max_\theta \Delta(Q(D_f), \mathcal{P}_f), \text{ where } x \sim \mathcal{P}_f, g_U^e(x) \sim Q(D_f). \tag{2}$$

In the above two equations, $\Delta(\cdot, \cdot)$ represents the discrepancy between two distributions. Eq. 1 and Eq. 2 describe the intuition that the unlearning extractor should attain the distribution of the remaining data but dissolve the distribution of forgetting data. We preserve the knowledge of the remaining data through Eq. 1, and treat the forgetting data as irrelevant data via Eq. 2. In this way, the forgetting data will be predicted based on the preserved knowledge instead of random guessing. We could use multi-objective solvers (Coello et al., 2002) for the optimization problem Eqs. 1 and 2. In this paper, considering the benefit of end-to-end models, we merge these two objectives into the following one for learning the optimal θ^*

$$\theta^* = \arg\min_\theta \Delta(Q(D_r), \mathcal{P}_r) - \Delta(Q(D_f), \mathcal{P}_f). \tag{3}$$

Since the size of D_f is limited, training new models on D_f becomes challenging. In addition, training new models on D_f can also be inefficient. We cannot directly have $\mathcal{P}_r$ and $\mathcal{P}_f$ and thus we first estimate them through the representation extractor of the g_D, i.e., g_D^e. This estimation is based on

the assumption that g_D can convey sufficient information on the data that it is trained on. However, g_D^e is a well-trained representation extractor, which cannot be used directly to approximate the data distribution. Moreover, there could be a discrepancy between the distribution of the forgetting data and the remaining data. Thus we need a model to catch the difference accurately.

With such a goal to mimic the distribution and capture the difference between D_r and D_f, we train two VAEs (Kingma & Welling, 2014) to approximate the distribution of representations of D_r and D_f. Specifically, we first train VAE h for the remaining data D_r. Note that instead of capturing the information with D_r, h captures the information of all training data D. This enables direct training of the VAE without the need to specify forgetting data. Such efficiency gain is crucial since computational resources and time are significant considerations. In addition, the number of forgetting data is always far less than the number of training data; thus h can sufficiently represent the distribution $\mathcal{P}_r$. Therefore,, we use the entire dataset D for training VAE h, leading to the following optimization function:

$$\arg\min_{h} \mathbb{E}_{z\sim\mathcal{N}(\mu_h,\sigma_h^2)} \log P_h(g_D^e(x_r)|z) + KL(\mathcal{N}(\mu_h,\sigma_h^2)||\mathcal{N}(0,\mathcal{I})), \tag{4}$$

where h outputs a representation for any $x_r \sim \mathcal{P}$, $\mathcal{N}(0,\mathcal{I})$ is the standard Gaussian distribution, $\mathcal{N}(\mu_h,\sigma_h^2)$ is the Gaussian distribution parameterized on μ_h and σ_h, and z is the reparameterized sample from the Gaussian distribution. Here μ_h and σ_h are the mean and standard deviation estimated by h on its encoding layer for $g_D^e(x), x \in D$.

In parallel, another VAE h_f is trained specifically for the representations of forgetting data D_f by

$$\arg\min_{h_f} \mathbb{E}_{z\sim\mathcal{N}(\mu_{h_f},\sigma_{h_f}^2)} \log P_{h_f}(g_D^e(x_f)|z) + KL(\mathcal{N}(\mu_{h_f},\sigma_{h_f}^2)||\mathcal{N}(0,\mathcal{I})), \tag{5}$$

where μ_{h_f} and σ_{h_f} are the mean and standard deviation estimated by h_f on its encoding layer for $g_D^e(x_f), x_f \in D_f$.

After we learned h, which captures the distribution of representations extracted by $g_D^e(x)$, we still need to have $Q(D_r)$, which is the distribution of representations extracted by $g_U^e(x)$ on the remaining data. As in the $\mathcal{P}$ case, the representation extractor itself cannot be used as distributions, and we still need another VAE to express that. Since we cannot learn a new VAE for the unknown g_U^e, we propose to use the VAE learned in Eq. 4 to describe the distribution on D_r. In this way, by fixing the VAE h, the objective in Eq. 1 to learn the g_U^e can be

$$\arg\min_{\theta} \mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)} \log P_h(g_U^e(x_r)|z) + KL(\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)||\mathcal{N}(0,\mathcal{I})), \tag{6}$$

where $\tilde{\mu}_h$ and $\tilde{\sigma}_h$ are the mean and standard deviation estimated by h on its encoding layer for $g_U^e(x_r), x_r \in D_r$. Another objective corresponding to Eq. 2 can be optimized by

$$\arg\max_{\theta} \mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2)} \log P_{h_f}(g_U^e(x_f)|z) + KL(\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2)||\mathcal{N}(0,\mathcal{I})), \tag{7}$$

where $\tilde{\mu}_{h_f}$ and $\tilde{\sigma}_{h_f}$ are the mean and standard deviation estimated by h_f on its encoding layer for $g_U^e(x_f), x_f \in D_f$.

We can then merge Eqs 6 and 7 in the same way as Eq. 3 into one overall objective

$$\begin{aligned}\theta^* = \arg\min_{\theta}(&\mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)} \log P_h(g_U^e(x_r)|z) - \mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2)} \log P_{h_f}(g_U^e(x_f)|z) + \\ &KL(\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)||\mathcal{N}(0,\mathcal{I})) - KL(\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2))||\mathcal{N}(0,\mathcal{I}))),\end{aligned} \tag{8}$$

The second part, $KL(\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)||\mathcal{N}(0,\mathcal{I})) - KL(\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2))||\mathcal{N}(0,\mathcal{I}))$, typically act as penalty terms, enforcing a regularization effect. By eliminating these terms, we aim to reduce the constraints on the model, thereby allowing a more flexible adjustment of the distribution during the unlearning process. Hence, we simplify the objective in Eq. 8 by removing the second part. Noticed that we only drop the two KL divergence terms for unlearning as follows and we use the complete Eq. 4 and Eq. 5 for the VAE training:

$$\theta^* \approx \arg\min_{\theta}(\mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_h,\tilde{\sigma}_h^2)} \log P_h(g_U^e(x_r)|z) - \mathbb{E}_{z\sim\mathcal{N}(\tilde{\mu}_{h_f},\tilde{\sigma}_{h_f}^2)} \log P_{h_f}(g_U^e(x_f)|z)). \tag{9}$$

In implementations, the log-likelihood is usually optimized by the L2 loss as the reconstruction loss of the input and output of the VAE. Furthermore, in the unlearning stage, the negative L_2 loss

can easily diverge and then lead to the breakdown of the whole deep model, which is known as *catastrophic unlearning* (Nguyen et al., 2020). Therefore, to avoid catastrophic unlearning, we propose a normalized form of L_2 loss and optimize θ^* by the extractor unlearning loss L_{UE}:

$$L_{UE} = \sum_{x \in X_r} \frac{\|g_U^e(x) - h(g_U^e(x))\|_2^2}{\|g_U^e(x) - h(g_U^e(x))\|_2^2 + 1} - \sum_{x \in X_f} \frac{\|g_U^e(x) - h_f(g_U^e(x))\|_2^2}{\|g_U^e(x) - h_f(g_U^e(x))\|_2^2 + 1}, \tag{10}$$

where X_r and X_f denote the inputs to the model of remaining data and forgetting data.

3.2 Representation Alignment

On the one hand, the extractor unlearning stage introduces two approximate estimations of the representation distribution $Q(D_r)$ and $Q(D_f)$. $g_U^e(x)$ necessitates further adjustment to mitigate the influences induced by this approximate estimation. On the other hand, after the extractor unlearning, the representation space of model g_U^e can not be aligned with the original classifier layers g_D^c. The adjusted model will suffer a performance drop in the predictions. Furthermore, due to the lack of supervision information y, the original classifier g_D^c cannot be adjusted to align with the representation space after the extractor unlearning to retain the prediction capability of the whole model. To maintain the prediction performances of updated models, one possible approach is to shift the representation space after the extractor unlearning to align with the original representation space on the remaining data. In this case, the adjusted model after the extractor unlearning can utilize the supervision knowledge of the original model without adjusting g_D^c using any label information. Therefore, we propose the representation alignment loss function and optimize g_U^e by minimizing the proposed loss:

$$L_{RA} = \sum_{x \in X_r} \log\left(\frac{\exp(simloss(g_U^e(x), g_D^e(x)))}{\sum_{\hat{x} \in X_f} \exp(simloss(g_U^e(\hat{x}), g_D^e(\hat{x}))/\tau)}\right), \tag{11}$$

where τ is a hyperparameter. As for the similarity loss function $simloss(\cdot, \cdot)$ in Eq. 11, we use the cosine similarity loss for implementations because of its normalization characteristics and huge success on the high dimensional representation learning (Chen et al., 2020). Different from the classical contrastive loss, the representation alignment loss compares the similarity of the representations of the models before and after the unlearning algorithm on the same data point. The representation alignment loss reduces the distance between the representations of the two models on the remaining data point and increases the dissimilarity between the representations of the two models on the forgetting data. In the implementation, we optimized the extractor unlearning loss L_{UE} and representation alignment loss L_{RA} alternately because the magnitudes of the two losses have large differences in different datasets.

For the training data with extra annotations, we add an additional supervised repairing stage to retain higher performances after the LAF if the labels of the part of the remaining data are available. In the implementation, we sample the same number of remaining data as the number of forgetting data D_f.

Algorithm 1 Supervision-free Unlearning Algorithm: LAF

Input:
The training data, D, consisting of the remaining data D_r and forgetting data D_f;
Two initialized VAEs, h and h_f;
The original model g_D with the representation extractor g_D^e;
Epochs for the unlearning, $Epoch_r$.

Output:
The updated model that removes the knowledge of the forgetting data, f_{θ^*}.

Train h and h_f on D and D_f by Eq. 4 and Eq. 5;
Fix h and h_f and set $g_U = g_D$;
for e_r in $range(Epoch_r)$:
    Sample from D_r and get samples as the same size as D_f;
    Update g_U^e by Eq. 10;
    Update g_U^e by Eq. 11;
if labels of partial remaining data are available:
    Repair g_U by supervised data for one epoch;
return g_U.

3.3 OVERALL ALGORITHM AND IMPLEMENTATION

We provide the pseudo-code of LAF in Algorithm 1. The LAF takes the inputs of training data D and inputs of forgetting data D_f to get the embeddings through the extractor g_U^e. Then these embeddings work as the inputs to train two VAEs using optimization functions (Eq. 4 and Eq. 5). The two VAEs are expected to learn the distribution of original training data and forgetting data. We then remove the knowledge of the forgetting data while preserving the knowledge of the remaining data via the extractor unlearning loss in Eq. 10. Subsequently, to maintain the performance of the post-unlearning model, we align the representation space of the remaining data with the original representation space via the representation alignment loss in Eq. 11. Furthermore, when the supervision information of the remaining data is available, the post-unlearning model can be further repaired.

4 EXPERIMENTS

In this section, we conduct experiments to answer three research questions to evaluate LAF:

- **RQ1**: How do the proposed LAF perform on the data removal, class removal, and noisy label removal tasks, as compared with the state-of-the-art unlearning methods?
- **RQ2**: Is the representation space after the implementation of LAF consistent with the space of retrained models?
- **RQ3**: How do L_{UE} and L_{RA} affect the performance of LAF?

4.1 SETTINGS

Datasets and Models. To validate the effectiveness of LAF, we conduct experiments on four datasets: **DIGITS** (MNIST) (LeCun, 1998), **FASHION** (Fashion MNIST)(Xiao et al., 2017), **CIFAR10** (Krizhevsky et al., 2009) and **SVHN** (Netzer et al., 2011). For the two MNIST datasets, we use a convolutional neural network (**CNN**) with two convolutional layers (LeCun et al., 1995) while for the two CIFAR datasets, we choose an **18-layer ResNet** backbone(He et al., 2016).

Baselines. Considering that the label-agnostic unlearning on deep models is still a research gap, we compare the performance of **LAF** (label-agnostic) and **LAF+R** (with supervised data for repairing) with seven fully supervised unlearning baselines including **Retrain** which are the golden standards from the retraining models, and six state-of-the-art unlearning works on deep models. The six baselines include four approximate unlearning works that have been published in the past year and one exact unlearning method: **NegGrad**, **Boundary** (Chen et al., 2023), **SISA** (Bourtoule et al., 2021), **Unroll** (Thudi et al., 2022), **T-S** (Chundawat et al., 2023), and **SCRUB** (Kurmanji et al., 2023). The detailed descriptions of these baselines and the implementation details are provided in Appendix 3. All the experiments on these baselines are conducted for five rounds of different random seeds.

Evaluation Setting. To validate the efficacy of the LAF, we establish experiments under three scenarios: (1) data removal, wherein 40% of training data labelled from 5 to 9 are randomly selected for removal; (2) class removal, where data from class 0 are designated as forgetting data for removal; (3) noisy label removal, in which 60% of training data labelled from 0 to 4 are randomly annotated as wrong labels and regarded as forgetting data for removal. For evaluations, we assess the performance of LAF in comparison to baseline methods using four metrics: $\textbf{Train}_\textbf{r}$, representing the prediction accuracy of the post-unlearning model on the remaining data; $\textbf{Train}_\textbf{f}$, denoting the prediction accuracy of the post-unlearning model on the forgetting data; **Test**, indicating the prediction accuracy of the post-unlearning model on the test data, which is further divided into $\textbf{Test}_r$ and $\textbf{Test}_f$ in the class removal task, representing test accuracy on the remaining and forgetting classes respectively; and **ASR**, which denotes the attack success rate of the membership inference attack (Shokri et al., 2017; Chen et al., 2021). For all the above four metrics, the closer value on $\textbf{Train}_\textbf{f}$ and **ASR** of the post-unlearning model to the retrained model indicates better knowledge removal performance while the closer value on **Test** to the retrained model indicates better knowledge preservation performance.

4.2 UNLEARNING PERFORMANCES

Table 1, 2, and 3 showcase the experiment results of the three distinct tasks: data removal, class removal, and noisy label removal. Upon comprehensive analysis of the experimental outcomes, it is

Table 1: Comparison results with other state-of-the-art methods in data removal (avg%±std%). The **bold** record indicates the best result and the underlined record indicates the second best result. The following tables use the same notations as this table.

Method	Data	Train_r	Train_f	Test	ASR	Data	Train_r	Train_f	Test	ASR
Retrain	DIGITS	99.56±0.05	98.84±0.10	99.04±0.10	49.80±0.53	FASHION	96.43±0.35	92.15±0.41	90.23±0.22	47.32±0.76
NegGrad		99.18±0.28	**98.86±0.41**	98.62±0.29	50.24±0.27		93.28±0.29	88.93±0.79	89.18±0.24	46.11±0.66
Boundary		97.65±1.02	95.36±2.50	96.63±1.35	46.83±2.09		56.28±4.69	46.58±4.04	53.00±3.66	48.03±1.41
SISA		99.06±0.12	98.60±0.07	98.92±0.02	33.78±0.01		91.98±0.19	90.76±0.07	89.92±0.24	33.33±0.02
Unrolling		**99.63±0.15**	99.34±0.33	**99.08±0.18**	46.50±0.60		89.83±0.30	83.88±0.65	81.21±0.34	47.69±0.50
T-S		94.01±0.77	93.09±2.73	93.72±1.03	47.82±0.64		82.96±1.14	86.77±2.13	82.46±1.24	45.90±1.30
SCRUB		99.28±0.04	99.03±0.12	98.95±0.08	46.68±0.80		90.88±0.09	88.62±0.28	88.75±0.11	45.23±0.94
LAF+R		99.47±0.14	99.35±0.65	98.89±0.10	**49.42±0.51**		**94.18±0.30**	95.00±1.62	**90.51±0.28**	**47.39±0.23**
LAF		98.03±0.68	97.29±1.43	97.30±0.78	47.92±0.84		91.54±2.67	**90.91±7.00**	87.53±3.26	46.89±0.88
Retrain	CIFAR10	84.03±0.20	78.05±1.34	87.20±0.65	57.48±0.88	SVHN	83.88±0.23	75.16±0.76	93.41±0.40	58.76±0.48
NegGrad		79.08±0.55	70.50±2.94	83.51±0.97	56.53±0.34		81.57±0.34	69.93±1.66	91.54±1.01	57.94±0.80
Boundary		54.73±1.32	18.73±3.33	51.23±2.55	62.79±0.95		64.85±2.06	28.62±1.89	73.07±1.96	89.17±3.29
SISA		66.78±0.10	53.12±0.74	54.30±0.05	37.53±0.02		82.48±0.17	67.79±0.34	82.57±0.83	50.19±0.38
Unrolling		57.82±1.66	30.91±2.86	61.31±1.51	56.97±1.27		70.98±1.87	47.68±2.72	83.27±0.48	55.39±0.98
T-S		70.31±2.32	72.17±3.91	77.71±2.02	54.64±1.58		78.36±0.13	73.50±0.62	90.60±0.61	55.77±1.42
SCRUB		29.16±1.07	0.47±0.93	25.18±0.78	54.03±0.64		22.32±0.04	0±0	19.59±0.07	65.26±1.24
LAF+R		**79.57±0.72**	**79.50±0.66**	**84.74±1.08**	57.74±0.62		**83.37±0.41**	**76.08±0.76**	**93.56±0.51**	**58.03±0.28**
LAF		78.03±1.55	73.30±3.96	82.22±2.57	**57.65±0.70**		81.63±0.49	76.11±1.49	92.32±0.58	57.85±0.89

Table 2: Comparison results with other state-of-the-art methods in class removal (avg%±std%)

Method	Data	Test_r	Test_f	ASR	Data	Test_r	Test_f	ASR
Retrain	DIGITS	98.81±0.15	0±0	26.49±1.41	FASHION	92.66±0.29	0±0	38.24±3.13
NegGrad		98.86±0.39	79.76±38.91	39.71±3.99		89.70±0.74	0.92±0.48	**37.64±2.32**
Boundary		98.59±0.23	95.63±5.27	38.51±4.25		86.04±1.41	1.68±0.69	39.06±2.57
SISA		**99.10±0.03**	**0±0**	50.12±0.23		**92.14±0.07**	**0±0**	50.00±0.02
Unrolling		97.05±1.25	79.63±39.00	40.16±2.52		88.72±0.86	0.40±0.24	40.61±1.87
T-S		61.31±40.52	0.16±0.19	35.83±11.47		91.84±0.31	21.16±7.24	24.82±0.61
SCRUB		99.02±0.06	95.41±3.35	32.04±2.75		91.40±0.24	0.42±0.36	33.84±0.60
LAF+R		99.05±0.04	0.10±0.13	**24.38±0.34**		91.95±0.29	0.08±0.08	35.00±2.16
LAF		98.03±0.20	0.26±0.11	52.25±2.61		89.73±0.37	2.43±1.46	31.35±0.71
Retrain	CIFAR10	87.01±0.64	0±0	67.76±1.58	SVHN	94.07±0.67	0±0	59.33±1.31
NegGrad		57.55±2.84	**0±0**	50.46±0.81		76.92±1.23	6.44±9.12	52.70±3.62
Boundary		83.33±1.36	1.00±0.66	61.22±3.37		90.59±1.32	15.99±5.01	60.88±2.61
SISA		73.52±0.37	**0±0**	50.12±0.02		**91.96±0.63**	**0±0**	**61.26±1.42**
Unrolling		84.26±1.53	**0±0**	**67.59±2.49**		92.48±0.64	93.31±2.56	57.20±1.64
T-S		86.47±0.91	6.21±5.07	44.95±4.43		92.73±0.64	10.29±5.14	49.62±0.97
SCRUB		32.93±0.84	**0±0**	50.59±1.42		20.99±0.31	**0±0**	66.13±1.98
LAF+R		**87.15±0.55**	0.15±0.09	58.16±1.08		**91.96±0.63**	**0±0**	**61.26±1.42**
LAF		82.38±0.97	2.15±1.96	50.46±1.96		85.80±1.14	0.33±0.51	56.33±0.49

observed that our proposed LAF-R method achieves the highest performance in 19 evaluations and secures the second-highest performance in 15 out of a total of 44 evaluations. In contrast, the SISA method manages to attain the highest performance in only 13 evaluations and the second-highest in 5 evaluations. Other methods under consideration lag in comparison to LAF-R and SISA. Furthermore, we observe that LAF-R consistently achieves either the best or second-best results in the tasks of data removal and noisy label removal, particularly under the metric of **ASR**. This suggests that the proposed LAF-R stands as a highly reliable unlearning algorithm in countering membership inference attacks. However, a limitation is noted in the class removal task; while LAF-R consistently maintains the highest test data accuracy for the remaining class, it falls short in sufficiently removing the information of the forgetting class. This can be due to the lack of label information, which can compel a shift in the prediction results of the forgetting class to other classes in unlearning (Tarun et al., 2023; Chundawat et al., 2023; Kurmanji et al., 2023).

Regarding the performance of the proposed LAF, it demonstrates comparable results in data removal and noisy label removal tasks, although it exhibits weaker performance in the class removal task. In Table. 1, LAF attains the best and the second best $\textbf{Train}_\textbf{f}$ on Fashion and SVHN. It can achieve the best **ASR** on CIFAR10, and the second best **ASR**s on the DIGITS dataset. Moreover, in all evaluations excluding those on the DIGITS dataset, LAF consistently ranks within the top 5 performances. The suboptimal results on the DIGITS dataset can primarily be attributed to the excessive removal of information of the forgetting data, subsequently impacting the performance of the remaining data. In the class removal task, as previously noted, the label-agnostic approach exhibits shortcomings when compared to supervised repairing (LAF-R) and other supervised unlearning methods. In the noisy label removal task, LAF further demonstrates its ability to mitigate the effects of noisy labels and enhance prediction accuracy, securing top-5 rankings in all accuracy evaluations. Furthermore, the

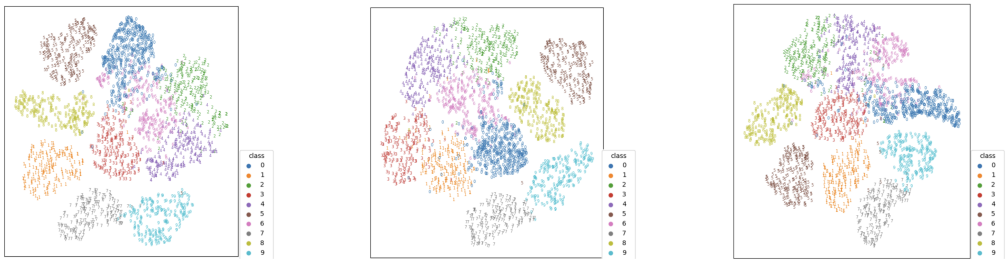
efficacy of the noisy label removal tasks also supports LAF can realize unlearning on low-quality representation extractor maintaining the prediction ability.

Table 3: Comparison results with other state-of-the-art methods in noisy label removal (avg%±std%)

Method	Data	Train_r	Train_f	Test	ASR	Data	Train_r	Train_f	Test	ASR
Retrain	DIGITS	99.75±0.12	0.17±0.01	98.83±0.05	39.26±0.01	FASHION	97.04±0.83	2.16±0.06	88.15±0.45	37.65±1.88
NegGrad		98.64±0.22	26.26±2.52	98.27±0.15	30.66±0.93		91.91±1.04	2.82±0.43	85.96±0.85	32.39±1.61
Boundary		82.05±9.04	7.05±3.03	69.85±14.83	29.44±1.14		72.82±6.71	11.21±1.79	54.54±10.37	30.58±1.19
SISA		**98.92±4.80**	1.50±0.04	**98.80±0.08**	24.64±0.02		92.22±5.95	**1.69±0.06**	**88.90±0.01**	25.00±0.06
Unrolling		67.86±0.26	0.43±0.09	97.31±0.55	31.35±1.10		61.73±1.83	3.76±0.83	80.02±3.85	33.97±1.31
T-S		90.71±3.52	3.60±1.11	83.85±5.69	27.05±1.76		85.56±3.13	5.64±1.32	74.19±5.23	28.86±0.78
SCRUB		97.27±0.39	0.74±0.18	96.31±0.63	31.48±1.08		87.29±1.35	4.29±0.50	79.41±2.28	33.32±0.26
LAF+R		98.87±0.19	**0.23±0.06**	98.45±0.23	**35.98±1.41**		**93.42±0.44**	2.06±0.20	87.71±0.36	**34.33±0.32**
LAF		96.46±0.67	2.70±0.59	91.48±1.49	18.51±0.57		92.32±0.66	4.80±0.71	81.21±1.22	22.36±0.72
Retrain	CIFAR10	73.33±0.89	7.74±0.23	64.74±1.26	57.04±0.99	SVHN	82.46±0.15	2.37±0.23	93.38±0.35	59.55±1.22
NegGrad		40.35±5.35	8.91±2.09	29.97±4.18	**55.98±0.37**		18.48±2.68	2.48±1.21	17.46±6.08	**58.25±2.05**
Boundary		42.69±3.44	8.23±1.35	33.57±2.04	54.81±2.17		44.27±1.43	7.86±0.51	51.66±1.43	58.15±1.12
SISA		**69.17±0.11**	**6.75±1.01**	**52.59±0.14**	28.62±0.02		**80.17±0.13**	**2.45±0.31**	80.02±0.07	44.84±0.04
Unrolling		32.81±4.34	8.88±2.42	32.01±3.87	53.86±1.26		29.71±3.00	10.52±0.99	32.33±5.03	53.61±0.58
T-S		57.50±2.38	10.97±0.83	45.92±4.81	50.57±1.11		75.45±0.33	4.27±0.29	83.87±0.56	51.16±2.01
SCRUB		51.84±1.00	10.70±0.41	38.06±0.37	52.38±1.57		59.89±1.66	5.10±0.44	66.80±4.96	57.22±0.90
LAF+R		60.49±1.71	9.33±0.35	51.73±2.27	54.49±1.04		79.39±0.27	2.85±0.17	**90.51±0.50**	55.09±1.64
LAF		57.44±1.11	10.60±0.20	47.57±0.63	53.18±0.68		77.87±0.35	3.59±0.20	89.33±0.32	51.50±1.17

4.3 Representation Space Visualization

Figure 1(a) and (c) present the visualized distributions of the representations before and after unlearning on the T-shirt class (blue points) and Figure 1(b) shows the representation distributions in the retrained model. In Figure 1(a), the forgetting data of the T-shirt class has a few intersecting distributions with the Shirt class (pink symbols) in the decision boundaries of the two classes. However, in the representation distributions of the retrained model, the cluster of the T-shirt shifts closer to the cluster of the Shirt, resulting in a greater overlap of data points between the T-shirt and Shirt classes. In the representation distributions of the post-unlearning model, which is shown in Figure 1(b), the clusters of the T-shirt consist of two segments. The first segment lies in the decision boundaries of the Dress and Shirt classes because the data of the T-shirt are easily misclassified as these two classes. The second segment near-completely overlaps with the data of the Shirt class, which is consistent with the representation distributions in the retrained model.



(a) Representation distributions before unlearning

(b) Representation distributions after retraining

(c) Representation distributions after unlearning

Figure 1: Representation distributions in the class removal task on the FASHION dataset. The blue number 0 stand for the forgetting data while the other numbers denotes the remaining data. The colour corresponding to each class is shown in the legends.

4.4 Ablation Study

Table 4, 5, and 6 delineate the results of the ablation study focusing on the impacts of losses L_{UE} and L_{RA} across three unlearning tasks. L_{UE} is proposed for extractor unlearning and L_{UE} is formulated for representation alignment to maintain the model's prediction performance. Therefore, it is anticipated that in the absence of L_{UE}, the post-unlearning model would exhibit inadequacy in removing forgetting data and without L_{RA}, the post-unlearning model will suffer degradation in the prediction performances.

The ablation study results corroborate these expectations. Firstly, in all three tables, the absence of optimization on L_{RA} for alignment with classifiers results in substantial performance degradation in both remaining data accuracy and test accuracy. This is particularly pronounced in class removal tasks. Additionally, in Table 5, the models lacking L_{UE} achieve significantly higher $\mathbf{Test_f}$ on the DIGITS, FASHION, and CIFAR10 datasets, indicating the ability to further remove the information of forgetting data. On the SVHN datasets, although the $\mathbf{Test_f}$ will be lower without L_{UE}, there is a notable decline in performance on the remaining data. In Table 4 and 6 where the forgetting data are randomly selected, the forgetting data distribution and the remaining data distribution are similar. Therefore, L_{UE} will have relatively minor influences on the unlearning process and the results without L_{UE} are close to the LAF results in the evaluations on the forgetting data.

Table 4: Ablation study results in data removal. 'None L1' denotes the unlearning without L_{UE} and 'None L2' denotes the unlearning without L_{RA}. The following tables take the same notations.

Method	Data	Train_r	Train_f	Test	ASR	Data	Train_r	Train_f	Test	ASR
Retrain	DIGITS	99.56±0.05	98.84±0.10	99.04±0.10	49.80±0.53	FASHION	96.43±0.35	92.15±0.41	90.23±0.22	47.32±0.76
None L1		**99.41±0.10**	**99.09±0.45**	**98.81±0.19**	47.02±1.39		87.30±2.77	77.08±7.43	81.44±3.45	46.05±0.49
None L2		19.02±1.93	38.49±15.16	22.06±4.38	44.62±1.52		44.24±2.47	81.68±6.84	50.67±3.00	40.95±0.07
LAF		98.03±0.68	97.29±1.43	97.30±0.78	**47.92±0.84**		**91.54±2.67**	**90.91±7.00**	**87.53±3.26**	**46.89±0.88**
Retrain	CIFAR10	84.03±0.20	78.05±1.34	87.20±0.65	57.48±0	SVHN	83.88±0.23	75.16±0.76	93.41±0.40	58.76±0.48
None L1		78.13±1.28	72.96±3.22	82.12±2.21	56.98±0.79		81.37±0.31	71.45±1.26	91.41±0.77	57.10±0.60
None L2		**78.62±0.80**	**80.62±1.59**	**84.67±0.56**	56.22±0.92		30.52±2.21	28.84±5.46	40.76±2.48	61.81±1.12
LAF		78.03±1.55	73.30±3.96	82.22±2.57	**57.65±0.70**		**81.63±0.49**	**76.11±1.49**	**92.32±0.58**	**57.85±0.89**

Table 5: Ablation study results in class removal.

Method	Data	Test_r	Test_f	ASR	Data	Test_r	Test_f	ASR
Retrain	DIGITS	98.81±0.15	0±0	26.49±1.41	FASHION	92.66±0.29	0±0	38.24±3.13
None L1		**98.88±0.09**	0.41±0.25	24.25±0.70		91.39±0.52	8.65±2.10	29.48±0.91
None L2		13.97±1.05	62.04±37.18	**26.37±0.92**		9.15±1.64	**1.53±1.47**	31.28±0.84
LAF		98.03±0.68	**0.26±0.11**	52.25±2.61		**91.54±2.67**	2.46±1.46	**31.35±0.71**
Retrain	CIFAR10	86.01±0.64	0±0	67.76±1.58	SVHN	94.07±0.67	0±0	59.33±1.31
None L1		7.63±2.22	34.93±21.11	**57.43±3.96**		61.13±4.78	**0.17±0.29**	61.36±9.89
None L2		33.02±3.61	3.60±1.32	53.33±4.23		9.54±0.54	2.19±3.09	**60.45±2.57**
LAF		**82.38±0.97**	**2.15±1.96**	50.46±1.96		**85.80±1.14**	0.33±0.51	56.33±0.49

Table 6: Ablation study results in noisy label removal.

Method	Data	Train_r	Train_f	Test	ASR	Data	Train_r	Train_f	Test	ASR
Retrain	DIGITS	99.75±0.12	0.17±0.01	98.83±0.05	**39.26±0.01**	FASHION	97.04±0.83	2.16±0.06	88.15±0.45	37.65±1.88
None L1		90.86±1.00	3.74±0.29	84.96±1.58	29.28±0.55		86.36±1.17	5.44±0.72	75.46±2.56	30.34±0.78
None L2		11.12±2.59	11.19±1.39	10.85±3.83	28.75±1.48		9.47±5.32	11.78±1.10	8.16±3.40	**32.41±1.00**
LAF		**96.46±0.67**	**2.70±0.59**	**91.48±1.49**	18.51±0.57		**92.32±0.66**	**4.80±0.71**	**81.21±1.22**	22.36±0.72
Retrain	CIFAR10	73.33±0.89	7.74±0.23	64.74±1.26	57.04±0.99	SVHN	82.46±0.15	2.37±0.23	93.38±0.35	59.55±1.22
None L1		57.44±1.10	10.61±0.21	47.57±0.63	53.41±0.52		**78.05±0.25**	**3.56±0.27**	89.10±0.54	51.14±0.65
None L2		54.31±2.98	10.65±0.24	46.86±2.49	**54.16±1.25**		34.63±3.33	5.41±1.16	43.63±8.15	**59.24±1.80**
LAF		**57.44±1.11**	**10.60±0.20**	**47.57±0.63**	53.18±0.68		77.87±0.35	3.59±0.20	**89.33±0.32**	51.50±1.17

5 CONCLUSION

In this study, addressing the imperative requirements for unlearning on the label-agnostic datasets, we introduce the Label-Agnostic Forgetting (LAF) framework. This framework is meticulously designed to eliminate the knowledge of the forgetting data distribution, while concurrently maintaining the knowledge of the remaining data at the representational level. Firstly, we employ two VAEs to model the distributions of both training and unlearning data, subsequently introducing a novel extractor unlearning loss to remove the knowledge of the forgetting data. Secondly, we introduce an additional representation alignment loss, intending to align the distributions of the remaining data representations with those preserved in the original model. Finally, if the annotations of any subset of remaining data are available, we proceed to update the entire model through supervised repairing, to further preserve the information of remaining data. The experiment results demonstrate the advantages of the LAF with supervised repairing (LAF+R), in comparison to baseline methodologies. Additionally, the findings also demonstrate the comparable efficacy of LAF without supervisory information, compared to other supervised unlearning approaches. The experiments also shed light on certain limitations of LAF, including the insufficient removal of the forgetting class in the class removal tasks, and the low efficiency compared with other supervised unlearning works. These observed limitations delineate prospective directions for future enhancements and refinements.

ACKNOWLEDGMENTS

The authors thank the NVIDIA Academic Hardware Grant Program for supporting their experiments, the UQ Cyber Security Fund (2021-R3 Weitong) for Shen, the Australian Research Council (DE230101116) for Xu, and the CMS Research Grants (15131570), Adelaide Nottingham Alliance Seed Fund Project (15133470), and ARC (DP240103070) for Chen.

REFERENCES

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *SP*, 2021.

Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In *CCS*, 2021.

Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *CVPR*, 2023.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan S. Kankanhalli. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In *AAAI*, 2023.

Carlos Artemio Coello Coello, David A. van Veldhuizen, and Gary B. Lamont. *Evolutionary algorithms for solving multi-objective problems*, volume 5 of *Genetic algorithms and evolutionary computation*. 2002.

Lydia de la Torre. A guide to the california consumer privacy act of 2018. *Available at SSRN 3275571*, 2018.

Jimmy Z. Di, Jack Douglas, Jayadev Acharya, Gautam Kamath, and Ayush Sekhari. Hidden poison: Machine unlearning enables camouflaged poisoning attacks. *Arxiv*, 2212.10717, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

Junyaup Kim and Simon S. Woo. Efficient two-stage model retraining for machine unlearning. In *CVPR*, 2022.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *https://www.cs.toronto.edu/ kriz/cifar.html*, 2009.

Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. Towards unbounded machine unlearning. *Arxiv*, 2302.09880, 2023.

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 1995.

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.

Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Variational bayesian unlearning. In *NeurIPS*, 2020.

Thanh Tam Nguyen, Thanh Trung Huynh, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning. *Arxiv*, 2209.02299, 2022.

Pierre Nodet, Vincent Lemaire, Alexis Bondu, Antoine Cornuéjols, and Adam Ouorou. From weakly supervised learning to biquality learning, a brief introduction. *Arxiv*, 2012.09632, 2020.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *SP*, 2017.

Ayush K. Tarun, Vikram S. Chundawat, Murari Mandal, and Mohan S. Kankanhalli. Fast yet effective machine unlearning. *TNNLS*, 2023.

Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling SGD: understanding factors influencing machine unlearning. In *EuroS&P*, 2022.

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

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *Arxiv*, 1708.07747, 2017.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. Machine unlearning: A survey. *ACM Comput. Surv.*, 2024.

Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *IEEE TKDE*, 35, 2023.

Peng-Fei Zhang, Guangdong Bai, Zi Huang, and Xin-Shun Xu. Machine unlearning for image retrieval: A generative scrubbing approach. In *MM*, 2022.