UNLEASHING MASK: EXPLORE THE INTRINSIC OUT-OF-DISTRIBUTION DETECTION CAPABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

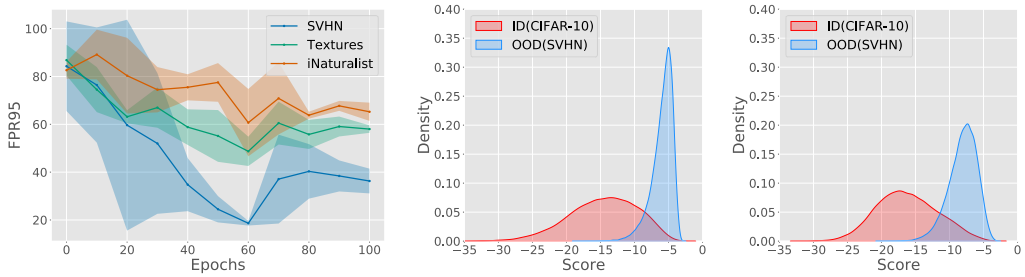
Out-of-distribution (OOD) detection is an important aspect for safely deploying machine learning models in real-world applications. Previous approaches either design better scoring functions or utilize the knowledge of outliers to equip the well-trained models with the ability of OOD detection. However, few of them explore to excavate the intrinsic OOD detection capability of a given model. In this work, we discover the existence of an intermediate stage of a model trained on in-distribution data having higher OOD detection performance than that of its final stage across different settings and further identify the critical attribution to be learning with *atypical samples*. Based on such empirical insights, we propose a new method, Unleashing Mask (UM), to reveal the once-covered detection capability of a given model. To be specific, we utilize the mask to figure out the memorized *atypical samples* and fine-tune the model to forget them. Extensive experiments have been conducted to characterize and verify the effectiveness of our method.

1 INTRODUCTION

Out-of-distribution (OOD) detection has drawn increasing attention when deploying machine learning models into the open-world scenarios (Nguyen et al., 2015; Lee et al., 2018a). Since the test samples can naturally arise from a label-different distribution, identifying OOD inputs is important, especially for those safety-critical applications like autonomous driving and medical intelligence. Previous studies focus on designing a series of scoring functions (Hendrycks & Gimpel, 2017b; Liang et al., 2018; Lee et al., 2018a; Liu et al., 2020; Sun et al., 2021; 2022) for OOD uncertainty estimation or finetuning with auxiliary outlier data to better distinguish the OOD inputs (Hendrycks et al., 2019c; Tack et al., 2020; Mohseni et al., 2020; Sehwag et al., 2021; Wei et al., 2022; Ming et al., 2022).

Despite the promising results achieved by previous methods (Hendrycks & Gimpel, 2017a; Hendrycks et al., 2019c; Liang et al., 2018; Liu et al., 2020; Ming et al., 2022), little attention is paid to considering whether the well-trained given model is the most appropriate for OOD detection. In general, models deployed for various applications have different targets (e.g., multi-class classification) (Goodfellow et al., 2016) instead of OOD detection (Nguyen et al., 2015; Lee et al., 2018a). However, most representative score functions, e.g., MSP (Hendrycks et al., 2019c), ODIN(Liang et al., 2018), and Energy (Liu et al., 2020), uniformly leverage the given models for OOD detection. Considering the target-oriented discrepancy, it arises a critical question: *does the well-trained given model have the optimal OOD detection capability?* If not, *how to find a more appropriate model for OOD detection?*

In this work, we start by revealing an important observation (as illustrated in Figure 1), i.e., there exists a historical training stage where the model has a higher OOD detection performance than the final well-trained one. Especially, it is a general existence across various setups, e.g., different OOD/ID datasets (Netzer et al., 2011; Van Horn et al., 2018; Cimpoi et al., 2014), learning rate schedules (Loshchilov & Hutter, 2017), and model structures (Huang et al., 2017; Zagoruyko & Komodakis, 2016). The empirical results of Figure 1 reflect the inconsistency between gaining better OOD detection capability (Nguyen et al., 2015) and pursuing better performance on ID data. We delve into the differences between the intermediate model and the final model by visualizing the misclassified examples. As shown in Figure 2, one possible attribution for covering the detection capability should be memorizing the *atypical samples* that are hard to learn for the model. Seeking zero error on those samples makes the model more confident on OOD data (see Figures 1(b) and 1(c)).



(a) Curves of FPR95 based on Energy (b) At Epoch 60 (c) At Epoch 100

Figure 1: (a) the curves of FPR95 based on the Energy score (Liu et al., 2020) across three different OOD datasets during the training on the `CIFAR-10` dataset. (b) comparison between ID and OOD distribution at Epoch 60. (c) comparison between ID and OOD distribution at Epoch 100. All the experiments testing for OOD detection performance have been conducted for multiple times. By backtracking the training phase, we can observe the existence of the model stage with better OOD detection capability using the Energy score to distinguish the OOD inputs. When zooming in the ID and OOD distributions at Epoch 60 and Epoch 100 respectively, it can be seen that, along with the training at the later stage, the overlap between them grows. More exploration can refer to Figure 2.

The above analysis inspires us to propose a new strategy, namely, *Unleashing Mask* (UM), to excavate the once-covered detection capability of a well-trained given model by alleviating the memorization of those atypical samples (as illustrated in Figure 3) of ID data. In general, we aim to backtrack its previous stage with a better OOD detection capability. To achieve this target, there are two essential issues: (1) *the model that is well-trained on ID data has already memorized some atypical samples*; (2) *how to forget those memorized atypical samples considering the given model?* Accordingly, our proposed UM contains two parts utilizing different insights to address the above problems. First, as atypical samples are more sensitive to the change of model parameters, we initialize a mask with the specific cutting rate to mine these samples with constructed discrepancy. Then, with the loss reference estimated by the mask, we conduct the constrained gradient ascent (i.e., Eq. 3) for model forgetting. It will encourage the model to finally stabilize around the expected stage. To avoid the severe sacrifice of the original task performance on ID data, we further propose UM Adopt Pruning (UMAP) which performs the tuning on the introduced mask with the newly designed objective.

For our proposed methods, we conduct extensive experiments to characterize and understand the working mechanism (in Section 4 and Appendix D). The comprehensive results accordingly demonstrate their rationality. We have verified the effectiveness of UM with a series of OOD detection benchmarks considering the two different ID datasets, i.e., CIFAR-10 and CIFAR-100. Under the various evaluation metrics, our UM, as well as UMAP, can indeed excavate the better OOD detection capability of given models and the averaged FPR95 can be reduced by a significant margin. Finally, a range of ablation studies and further discussions related to our proposed strategy has been provided. We summarize our main contributions as follows,

- Conceptually, we explore the OOD detection performance via a new perspective, i.e., backtracking the model training phase without regularizing by any auxiliary outliers, different from most previous works that start with the well-trained model on ID data.
- Empirically, we reveal the potential detection capability of the well-trained model. We observe the general existence of an intermediate stage where the model has more appropriate discriminative features that can be utilized for OOD detection.
- Technically, we introduce a new strategy, i.e., *Unleashing Mask*, to excavate the once-covered OOD detection capability of a given model. By introducing the mask, we estimate the loss constraint for forgetting the atypical samples and empower the detection performance.
- Experimentally, we conduct extensive explorations to verify the general effectiveness on improving the OOD detection performance of our methods. Using various ID and OOD benchmarks, we provide comprehensive results across different setups and further discussion.

2 BACKGROUND

In this section, we briefly introduce the preliminaries and related work about OOD detection.

2.1 PRELIMINARIES

We consider multi-class classification as the training task, where $\mathcal{X} \subset \mathbb{R}^d$ denotes the input space and $\mathcal{Y} = \{1, \dots, C\}$ denotes the label space. Considering the model deployed in the real world, the reliable classifier is expected to figure out the OOD input, which can be formulated as a binary classification problem. Given $\mathcal{P}$, the distribution over $\mathcal{X} \times \mathcal{Y}$, we consider D_{in} as the marginal distribution of $\mathcal{P}$ for $\mathcal{X}$, namely, the distribution of ID data. At test time, the environment can present a distribution D_{out} over $\mathcal{X}$ of OOD data. In general, the OOD distribution D_{out} is defined as an irrelevant distribution of which the label set has no intersection with $\mathcal{Y}$ Yang et al. (2021) and therefore should not be predicted by the model. The decision can be made with the threshold λ:

$$D_\lambda(x; f) = \begin{cases} \text{ID} & S(x) \geq \lambda \\ \text{OOD} & S(x) < \lambda \end{cases}, \tag{1}$$

Building upon the model $f \in \mathcal{H} : \mathcal{X} \to \mathbb{R}^c$ trained on ID data with the logit outputs, the goal of decision is to utilize the scoring function $S : \mathcal{X} \to \mathbb{R}$ to distinguish the inputs of D_{in} from that of D_{out} by $S(x)$. Typically, if the score value is larger than the threshold λ, the associated input x is classified as ID and vice versa. We consider several representative scoring functions designed for OOD detection, e.g., MSP (Hendrycks & Gimpel, 2017b), ODIN (Liang et al., 2018), and Energy (Liu et al., 2020). More detailed definitions and implementation can refer to Appendix A.

To mitigate the issue of over-confident predictions for (Hendrycks & Gimpel, 2017b; Liu et al., 2020) some OOD data, recent works (Hendrycks et al., 2019c; Tack et al., 2020) utilize the auxiliary unlabeled dataset to regularize the model behavior. Among them, one representative baseline is outlier exposure (OE) (Hendrycks et al., 2019c). OE can further improve the detection performance by making the model $f(\cdot)$ finetuned from a surrogate OOD distribution $D^{\text{s}}_{\text{out}}$, and its corresponding learning objective is defined as follows,

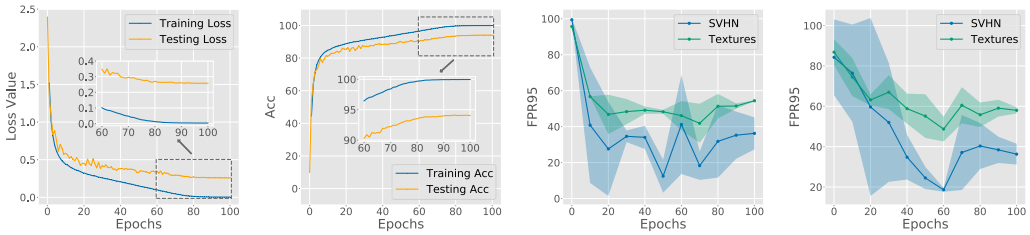
$$\mathcal{L}_f = \mathbb{E}_{D_{\text{in}}}\left[\ell_{\text{CE}}(f(x), y)\right] + \lambda \mathbb{E}_{D^{\text{s}}_{\text{out}}}\left[\ell_{\text{OE}}(f(x))\right], \tag{2}$$

where λ is the balancing parameter, $\ell_{\text{CE}}(\cdot)$ is the Cross-Entropy (CE) loss, and $\ell_{\text{OE}}(\cdot)$ is the Kullback-Leibler divergence to the uniform distribution, which can be written as $\ell_{\text{OE}}(h(\boldsymbol{x})) = -\sum_k \texttt{softmax}_k\, f(x)/C$, where $\texttt{softmax}_k(\cdot)$ denotes the k-th element of a softmax output. The OE loss $\ell_{\text{OE}}(\cdot)$ is designed for model regularization, making the model learn from surrogate OOD inputs to return low-confident predictions. The general formulation of Eq 2 is also adopted in other related works for designing better tuning objectives that use different auxiliary outlier data.

Although previous works show promising results via designing scoring functions or regularizing models based on the model f trained on ID data, few of them investigated the original detection capability of the well-trained given model. In this work, we introduce the layer-wise mask m (Han et al., 2016; Ramanujan et al., 2020) to mine the atypical samples that memorized by the model. Accordingly, the decision can be written as $D(x; m \odot f)$, and the output of masked model is $m \odot f(x)$.

2.2 RELATED WORK

Out-of-distribution Detection without auxiliary data. (Hendrycks & Gimpel, 2017a) formally shed light on out-of-distribution detection, proposing to use softmax prediction probability as a baseline which is demonstrated to be unsuitable for OOD detection. Subsequent works keep focusing on designing post-hoc metrics to distinguish ID samples from OOD samples, among which ODIN (Liang et al., 2018) introduces small perturbations into input images to facilitate the separation of softmax score, Mahalanobis distance-based confidence score (Lee et al., 2018b) exploits the feature space by obtaining conditional Gaussian distributions, energy-based score (Liu et al., 2020) aligns better with the probability density. Besides designing score functions, many other works pay attention to various aspects to enhance the OOD detection such that LogitNorm (Wei et al., 2022) produces confidence scores by training with a constant vector norm on the logits, and DICE (Sun & Li, 2022) reduces the variance of the output distribution by leveraging the sparsification of the model.



(a) Training/Testing Loss (b) Training/Testing Acc. (c) ODIN Score (d) Energy Score

(e) Correctly classified samples at Epoch 60 (f) Mis-classified samples learned after Epoch 60

Figure 2: (a): training/testing loss on ID data; (b) training/testing accuracy on ID data; (c): curves of FPR95 based on ODIN score; (d): curves of FPR95 based on Energy score; (e): visualization of correct classified samples at Epoch 60 (f): visualization of misclassified samples at Epoch 60. We investigate the once-covered OOD detection capability by checking the model behavior during its training phase. We take a closer look at the corresponding training and testing performance with the OOD detection capability indicated by two different scores. Through comparison, we find that achieving a reasonably small loss value (at round Epoch 60) on ID data is enough for OOD detection. However, continually optimizing on those atypical samples can impair the detection performance.

Out-of-distribution Detection with auxiliary data. Another promising direction towards OOD detection involves the auxiliary outliers for model regularization. On one hand, some works generate virtual outliers such that Lee et al. (2018a) uses generative adversarial networks to generate boundary samples, VOS (Du et al., 2022a) regularizes the decision boundary by adaptively sampling virtual outliers from the low-likelihood region. On the other hand, other works tend to exploit information from natural outliers, such that outlier exposure (OE) is introduced by Hendrycks et al. (2019b), given that diverse data are available in enormous quantities. (Yu & Aizawa, 2019) train an additional "head" and maximizes the discrepancy of decision boundaries of the two heads to detect OOD samples. Energy-bounded learning (Liu et al., 2020) fine-tunes the neural network to widen the energy gap by adding an energy loss term to the objective. Some other works also highlight the sampling strategy, such that ATOM (Chen et al., 2021) greedily utilizes informative auxiliary data to tighten the decision boundary for OOD detection, and POEM (Ming et al., 2022) adopts Thompson sampling to contour the decision boundary precisely. The performance of training with outliers is usually superior to that without outliers, shown in many recent works (Mohseni et al., 2020; Liu et al., 2020; Fort et al., 2021; Sun et al., 2021; Sehwag et al., 2021; Yang et al., 2021; Chen et al., 2021; Salehi et al., 2021).

3 Proposed Method: Unleashing Mask

In this section, we introduce our new method, i.e., *Unleashing Mask* (UM), to reveal the potential OOD detection capability of the well-trained model. First, we present and discuss the important observation that inspires our methods (Section 3.1). Second, we provide the insights behind the two critical parts of our UM (Section 3.2). Lastly, we describe the framework and the learning objective of UM that incorporates the previous component, as well as its variant, i.e., UMAP (Section 3.3).

3.1 Once-covered OOD Detection Capability

First, we present the phenomenon of the inconsistent trend between a better OOD detection capability and smaller training error during training. Empirically, as shown in Figure 1, we plot the OOD detection performance during the model training after multiple runs of the experiments. Across three

different OOD testing datasets, we can observe the existence of a better detection capability using the index of FPR95 metric based on the Energy (Liu et al., 2020) or ODIN (Liang et al., 2018) score. The generality has also been verified using different learning schedules and model structures in our experimental sections. We further show the comparison of the ID/OOD distributions in Figures 1(b) and 1(c). To be specific, the statics of two distributions indicate that the gap between the ID and OOD data gets narrow as their overlap grows along with the training. After Epoch 60, although the model becomes more confident on ID data which satisfies a part of the calibration target (Hendrycks et al., 2019a), its predictions on the OOD data also become more confident which is unexpected. Without seeing any auxiliary outliers in training, it motivates us to explore how the model achieves that.

We take a closer look at the model behaviors in Figure 2, where we check its corresponding training/testing loss and accuracy. We find that the training loss has reached a reasonably small value at Epoch 60 where its detection performance also achieves a satisfactory level. However, if we further minimize the training loss, the trend of the FPR95 curve shows almost opposite directions with both training and testing loss or accuracy (see Figures 1(a), 2(b), and 2(c)). Accordingly, we extract those samples that were learned by the model at this period. As shown in Figures 2(e) and 2(f), the misclassified samples learned after Epoch 60 present much atypical semantic features than those well-learned samples. As deep neural networks tend to first learn the data with typical features (Arpit et al., 2017), we attribute the inconsistent trend to memorizing those atypical data at the later stage.

3.2 Unleashing the intrinsic detection power

In general, the models that are developed for the original classification tasks are always seeking better performance (i.e., higher testing accuracy and lower training loss) in practice. However, the inconsistent trend revealed before indicates that the intrinsic OOD detection capability maybe once-covered during the training phase. It gives us a chance to unleash the potential detection power only considering the ID data during training. To this end, we have two important issues that need to address: (1) *the model that is well-trained on ID data may already memorized some atypical samples which can not be figured out*; (2) *how to forget those atypical samples considering the given model?*

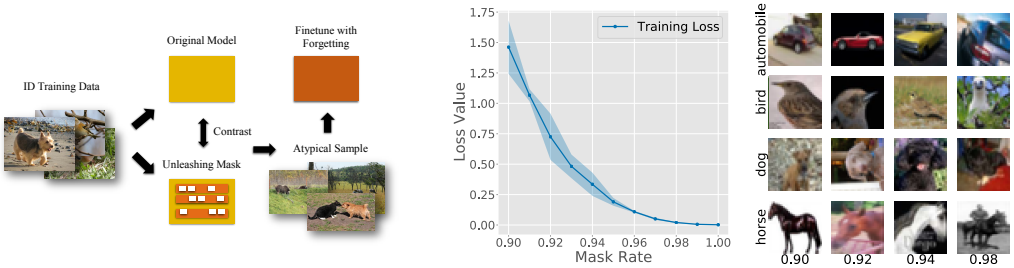
Mining the atypical samples with constructed discrepancy. According to Figures 2(a) and 2(b), both training accuracy and loss provide limited information that can differentiate the typical and atypical data. Inspired by the learning dynamics (Goodfellow et al., 2016; Arpit et al., 2017) of deep neural networks and the pathway conjecture (Barham et al., 2022) for inference, we try to manually construct the parameter discrepancy to mine the atypical samples from a well-trained model. To be specific, we introduce a novel layer-wise mask to achieve the goal. In Figure 3(c), we provide empirical evidence to show that we can figure out the atypical samples via enlarging the mask rate. Utilizing the masked output for loss computation, the atypical samples can be better differentiated.

Forgetting the atypical samples with gradient ascent. As the training loss achieves zero at the final stage of the given model, we need extra optimization signals to forget those memorized atypical samples. Considering the previous consistent trend before the potential optimal stage (e.g., before Epoch 60 in Figure 1(a)), the optimization signal also needs to control the model update not to be too greedy to drop the discriminative features for OOD detection. Starting with the given model, we can employ the gradient ascent (Sorg et al., 2010; Ishida et al., 2020) to forget the targeted samples, while the tuning phase should also prevent further updates if the model can achieve the expected stage.

3.3 Method Realization

Based on previous insights, we present our overall framework as well as the learning objective of the proposed *Unleashing Mask* for OOD detection. Lastly, we discuss its compatibility with either the fundamental scoring functions or the outlier exposure approaches utilizing auxiliary outlier data.

Framework. As illustrated in Figure 3(a), our framework consists of two critical components for uncovering the intrinsic OOD detection capability: (1) the initialized mask with a specific pruning rate for constructing the output discrepancy with the original model; (2) the fine-tuning procedure for alleviating the memorization of atypical samples. The overall workflow starts with obtaining the loss value of misclassifying those atypical samples and then conducts tuning with the model to forget.



(a) Illustration of Unleashing Mask (UM) (b) Mask Rate w.r.t. Loss (c) Mis-classified data

Figure 3: (a): a brief illustration of the proposed Unleashing Mask (UM); (b): the mask rate w.r.t. loss value using the masked outputs; (c) examples for misclassified samples after masking the original well-trained model. As for our framework, given a well-trained model, we initialize an extra mask for mining the atypical samples that sensitive to the changes in model parameters. Then fine-tune the original model or adopt pruning with the estimated forgetting threshold, i.e., the loss value estimated by the UM. The final model can serve as the base of various score functions to utilize the discriminative features and also as the new initialization of fine-tuning with the auxiliary outliers.

Objective for forgetting. Based on our framework, we introduce our learning objective for forgetting as follows,

$$\min \mathcal{L}_{\text{UM}} = \min_{m_\delta \in [0,1]^n} |\ell_{\text{CE}}(f) - \widehat{\ell}_{\text{CE}}(m_\delta \odot f)| + \widehat{\ell}_{\text{CE}}(m_\delta \odot f), \tag{3}$$

where m_δ is our proposed layer-wise mask with the pruning rate δ, ℓ_{CE} is the CE loss, and $\widehat{\ell}_{\text{CE}}$ is the averaged CE loss over the ID training data, $|\cdot|$ indicate the computation for absolute value and $m_\delta \odot f$ indicated the masked output. Concretely, the well-trained model will start to optimize itself again if it memorized the atypical samples and achieved almost zero loss value. We provide a positive gradient signal when the current loss value is lower than the estimated one and vice versa. The mode is expected to finally stabilize around the stage that can forget those atypical samples. Considering the potential negative effect on the original task performance when conducting tuning for forgetting, we further propose a variant of UM Adopting Pruning, i.e., UMAP, to conduct tuning based on the masked output (e.g., replace $\ell_{\text{CE}}(f)$ to $\ell_{\text{CE}}(m \odot f)$ in Eq 3) with the objective. We summarize the detailed algorithmic realizations of UM (in Algorithm 1) and UMAP (in Algorithm 2) in Appendix B.

Compatible to other methods. As we explore the original OOD detection capability of the well-trained model, it is orthogonal and compatible with those promising methods that equip the given model with better detection ability. To be specific, through our proposed methods, we reveal the once-covered OOD detection capability via tuning the original model towards its intermediate training stage. The discriminative feature learned at that stage can be utilized by different scoring functions (Huang et al., 2021; Sun & Li, 2022; Wei et al., 2022), like ODIN (Liang et al., 2018) adopted in Figure 2(c). For those methods (Hendrycks et al., 2019a; Liu et al., 2020; Ming et al., 2022) utilizing the auxiliary outliers to regularize the model, our fine-tuned model obtained by UM and UMAP can also serve as their starting point or adjustment. As our strategy does not require any auxiliary outlier data to be involved in training, adjusting the model using ID data during its developing phase is practical.

4 Experiments

In this section, we present the performance comparison of the proposed method in the OOD detection scenario. Specifically, we verify the effectiveness of our UM and UMAP with two mainstreams of OOD detection approaches: (i) fundamental scoring function methods; (ii) outlier exposure methods involving auxiliary samples. To better understand and characterize our proposed method, we further conduct extensive explorations on the ablation study and provide the corresponding discussion on each sub-aspect considered in our work. More details and additional results can also refer to Appendix D.

4.1 EXPERIMENTAL SETUPS

Datasets. Following the common benchmarks used in previous work (Liu et al., 2020; Ming et al., 2022), we adopt `CIFAR-10`, `CIFAR-100` (Krizhevsky, 2009) as our ID datasets. We use a series of different image datasets as the OOD datasets, namely `Textures` (Cimpoi et al., 2014), `Places365` (Zhou et al., 2017), `SUN` (Xiao et al., 2010), `LSUN` (Yu et al., 2015), and `iNaturalist` (Van Horn et al., 2018). We also use the other ID dataset as OOD dataset when training on a specific ID dataset, given that none of them shares any same classes (Yang et al., 2021). e.g. we treat `CIFAR-100` as the OOD dataset when training on `CIFAR-10` in our experiments for comparison.

Training details. We conduct all major experiments on DenseNet-101 (Huang et al., 2017) with training epochs fixed to 100. We also include experiment results on other types of models in the Appendix D. The models are trained using stochastic gradient descent (Kiefer & Wolfowitz, 1952) with Nesterov momentum (Duchi et al., 2011). We adopt Cosine Annealing (Loshchilov & Hutter, 2017) to schedule the learning rate which begins at 0.1. We set the momentum and weight decay to be 0.9 and 10^{-4} respectively throughout all experiments. The size of the mini-batch is 64 for both ID samples (when training and testing) and OOD samples (when testing). We empirically determine the value of our proposed UM and UMAP by examining the training loss on the masked output. For `CIFAR-10` as ID datasets, the value of mask rate is 97.5% and the estimated loss constraint for forgetting is 0.1 for our tuning until the convergence; For `CIFAR-100`, the value of mask rate is 97% and the estimated loss constraint for forgetting is 1.2 for our tuning until the convergence.

Masking details. Specifically, we introduce the layer-wise mask with flexible rates for determining the masked percentage of the model parameters. In specific, the mask is added to each layer and its value is set to be 0 to mask the parameter or 1 to keep the parameter in inference. In the major experiments of our performance comparison, we do not consider the optimization process to update the mask as the general pruning. In the ablation, we consider the simple-to-implemented edge-popup score (Ramanujan et al., 2020) as our main strategy to explore the pruning effect. Other different pruning methods (Han et al., 2016; Dong et al., 2017) can also be adopted into the explorations.

Evaluation metrics. We employ the following three common metrics to evaluate the performance of OOD detection: (i) Area Under the Receiver Operating Characteristic curve (AUROC) (Davis & Goadrich, 2006) can be interpreted as the probability for a positive sample to have a higher discriminating score than a negative sample (Fawcett, 2006); (ii) Area Under the Precision-Recall curve (AUPR) (Manning & Schütze, 1999) is an ideal metric to adjust the extreme difference between positive and negative base rates; (iii) False Positive Rate (FPR) at 95% True Positive Rate (TPR) (Liang et al., 2018) indicates the probability for a negative sample to be misclassified as positive when the true positive rate is at 95%. We also include in-distribution testing accuracy (ID-ACC) to reflect the preservation level of the performance for the original classification task on ID data.

OOD detection baselines. We compare the proposed method with several competitive baselines in the two directions. Specifically, we adopt Maximum Softmax Probability (MSP) (Hendrycks & Gimpel, 2017a), ODIN (Liang et al., 2018), Mahalanobis score (Lee et al., 2018b), and Energy score (Liu et al., 2020) as scoring function baselines; We adopt OE (Hendrycks et al., 2019b), Energy-bounded learning (Liu et al., 2020), and POEM (Ming et al., 2022) as baselines with outliers. For all scoring function methods, we assume the accessibility of well-trained models. For all methods involving outliers, we constrain all major experiments to a fine-tuning scenario, which is more practical in real cases. Different from training a dual-task model at the very beginning, equipping deployed models with OOD Detection ability is a much more common circumstance, considering the millions of existential and running deep learning systems. We leave more details in Appendix A.

4.2 PERFORMANCE COMPARISON

In this part, we present the performance comparison with some representative baseline methods to demonstrate the effectiveness of our UM and UMAP. Our proposed UM is designed for excavating the potential OOD detection capability of the given model. Here we consider several scoring functions to compare the detection performance, and also some outlier exposure methods to further regularize

Table 1: Main Results (%). Comparison with competitive OOD detection baselines.

D_{in}	Method	AUROC↑	AUPR↑	FPR95↓	ID-ACC↑	w./w.o D_{aux}
CIFAR-10	MSP(Hendrycks & Gimpel, 2017a)	89.90 ± 0.30	91.48 ± 0.43	60.08 ± 0.76	94.01 ± 0.08	
	ODIN(Liang et al., 2018)	91.46 ± 0.56	91.67 ± 0.58	42.31 ± 1.38	94.01 ± 0.08	
	Mahalanobis(Lee et al., 2018b)	75.10 ± 1.04	72.32 ± 1.92	61.35 ± 1.25	94.01 ± 0.08	
	Energy(Liu et al., 2020)	92.07 ± 0.22	92.72 ± 0.39	42.69 ± 1.31	94.01 ± 0.08	
	Energy+UM (ours)	93.73 ± 0.36	94.27 ± 0.60	33.29 ± 1.70	92.80 ± 0.47	
	Energy+UMAP (ours)	$\mathbf{93.97 \pm 0.11}$	$\mathbf{94.38 \pm 0.06}$	$\mathbf{30.71 \pm 1.94}$	$\mathbf{94.01 \pm 0.08}$	
	OE(Hendrycks et al., 2019b)	97.07 ± 0.01	97.31 ± 0.05	13.80 ± 0.28	92.59 ± 0.32	✓
	Energy (w. D_{aux})(Liu et al., 2020)	94.58 ± 0.64	94.69 ± 0.65	18.79 ± 2.31	80.91 ± 3.13	✓
	POEM(Ming et al., 2022)	94.37 ± 0.07	94.51 ± 0.06	18.50 ± 0.33	77.24 ± 2.22	✓
	OE+UM (ours)	$\mathbf{97.60 \pm 0.03}$	$\mathbf{97.87 \pm 0.02}$	$\mathbf{11.22 \pm 0.16}$	93.66 ± 0.12	✓
	OE+UMAP (ours)	97.48 ± 0.01	97.74 ± 0.00	12.21 ± 0.09	$\mathbf{94.01 \pm 0.08}$	✓
CIFAR-100	MSP(Hendrycks & Gimpel, 2017a)	74.06 ± 0.69	75.37 ± 0.73	83.14 ± 0.87	74.86 ± 0.21	
	ODIN(Liang et al., 2018)	76.18 ± 0.14	76.49 ± 0.20	78.93 ± 0.31	74.86 ± 0.21	
	Mahalanobis(Lee et al., 2018b)	63.90 ± 1.91	64.31 ± 0.91	78.79 ± 0.50	74.86 ± 0.21	
	Energy(Liu et al., 2020)	$\mathbf{76.29 \pm 0.24}$	$\mathbf{77.06 \pm 0.55}$	78.46 ± 0.06	74.86 ± 0.21	
	Energy+UM (ours)	76.22 ± 0.42	76.39 ± 1.03	74.05 ± 0.55	64.55 ± 0.24	
	Energy+UMAP (ours)	75.57 ± 0.59	75.66 ± 0.07	$\mathbf{72.21 \pm 1.46}$	$\mathbf{74.86 \pm 0.21}$	
	OE(Hendrycks et al., 2019b)	90.55 ± 0.87	90.34 ± 0.94	34.73 ± 3.85	73.59 ± 0.30	✓
	Energy (w. D_{aux})(Liu et al., 2020)	88.92 ± 0.57	89.13 ± 0.56	37.90 ± 2.59	57.85 ± 2.65	✓
	POEM(Ming et al., 2022)	88.95 ± 0.54	88.94 ± 0.31	38.10 ± 1.30	56.18 ± 1.92	✓
	OE+UM (ours)	91.04 ± 0.11	$\mathbf{91.13 \pm 0.24}$	34.71 ± 0.81	$\mathbf{75.15 \pm 0.18}$	✓
	OE+UMAP (ours)	$\mathbf{91.10 \pm 0.16}$	90.99 ± 0.23	$\mathbf{33.62 \pm 0.26}$	74.76 ± 0.11	✓

Table 2: Fine-grained Results (%). Comparison on different OOD benchmark datasets.

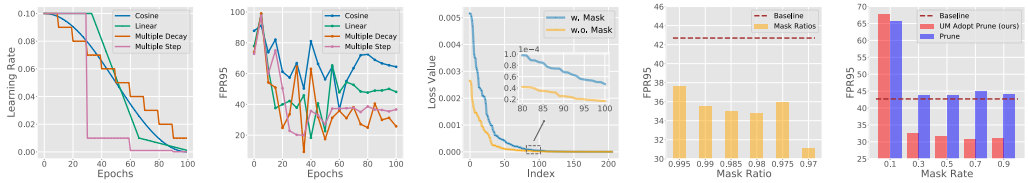
ID dataset	Method	OOD dataset					
		CIFAR-100		Textures		Places365	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10	MSP	66.43 ± 1.25	87.73 ± 0.02	65.20 ± 1.33	88.06 ± 0.61	61.34 ± 0.60	89.63 ± 0.15
	ODIN	55.31 ± 0.85	87.75 ± 0.37	53.11 ± 4.84	87.13 ± 2.04	43.77 ± 0.20	91.70 ± 0.30
	Mahalanobis	81.61 ± 0.96	$64.52 \pm 0..73$	$\mathbf{20.04 \pm 1.43}$	$\mathbf{94.38 \pm 0.78}$	86.21 ± 1.36	64.00 ± 1.21
	Energy	54.65 ± 1.24	$\mathbf{89.01 \pm 1.18}$	57.09 ± 3.52	87.51 ± 1.43	38.62 ± 1.64	93.03 ± 0.20
	Energy+UM (ours)	$\mathbf{54.62 \pm 1.16}$	88.30 ± 0.30	41.61 ± 3.67	91.31 ± 0.01	$\mathbf{30.85 \pm 0.58}$	$\mathbf{94.27 \pm 0.16}$
	Method	**SUN**		**LSUN**		**iNaturalist**	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
	MSP	60.27 ± 0.66	90.00 ± 0.24	36.43 ± 1.94	95.17 ± 0.32	67.53 ± 1.64	88.01 ± 0.82
	ODIN	41.14 ± 1.29	92.34 ± 0.62	5.16 ± 0.76	98.96 ± 0.09	54.41 ± 0.91	90.17 ± 0.19
	Mahalanobis	84.56 ± 1.51	66.41 ± 4.57	69.18 ± 3.52	66.41 ± 4.57	80.76 ± 2.48	71.77 ± 1.12
	Energy	36.73 ± 1.72	93.63 ± 0.34	6.25 ± 0.43	98.77 ± 0.07	59.11 ± 1.18	89.71 ± 0.06
	Energy+UM (ours)	$\mathbf{27.88 \pm 0.73}$	$\mathbf{94.83 \pm 0.11}$	$\mathbf{2.91 \pm 0.53}$	$\mathbf{99.22 \pm 0.11}$	$\mathbf{46.27 \pm 2.74}$	$\mathbf{92.75 \pm 0.80}$

the given model and boost the OOD detection ability. In each category, we choose one with the best detection performance to adopt UM/UMAP and check the detection results with the ID-ACC.

In Table 1, we summarize the results on different OOD test sets using different methods. Note that, here the evaluation results are obtained by averaging several OOD test datasets across multiple independent trials. For the scoring-based methods, our UM can further improve the overall detection performance by alleviating the memorization on atypical ID data, when the ID-ACC keeps comparable with the baseline. For the complex `CIFAR-100` dataset, our UMAP can be adopted as a practical way to empower the detection performance and simultaneously avoid affecting the original performance on ID data. As for those methods of second category (i.e., involving auxiliary outlier D_{aux} sampling from ImageNet), since we consider a practical workflow, i.e., fine-tuning, on the given model, OE achieves the best performance on the task. Due to the special optimization characteristic, Energy (w.D_{aux}) and POEM focus more on the energy loss on differentiating OOD data while performing not well on the preservation of ID-ACC. Without sacrificing much performance on ID data, OE with our UM can still achieve better detection performance. In Table 3, the fine-grained detection performance on each OOD testing set demonstrates the general effectiveness of UM. We have comprehensively verified the significant improvement (up to 18% reduced on averaged FPR95) on OOD detection of our methods across different setups, the complete results can refer to Tables 4 to 8 in Appendix D.

4.3 Ablation and Further Discussion

In this part, we conduct various explorations to provide a thorough understanding of our presented Unleashing Mask from different perspectives. To be specific, we first present the general existence of



(a) lr Schedule (b) FPR95 Curves (c) w./w.o. Mask (d) Masking Ratios (e) Pruning w. Mask

Figure 4: Ablation studies on different aspects. (a)-(b): exploring the existence of potential OOD detection capability with different learning rate schedules; (c) comparison of loss value using original output with masked output; (d) comparison of using original pruning with our proposed UMAP.

the once-covered intrinsic OOD detection capability across different settings. Second, we present the exploration about the mask, which helps us to characterize its effect on figuring out the atypical samples. Third, we provide further exploration on excavating the detection power via pruning.

General existence of once-covered OOD detection capability. In Figure 4(a) and Figure 4(b), we explore 4 learning rate schedules to demonstrate the general existence of once-covered OOD detection capability. To be specific, the OOD performance (indicated by FPR95) is evaluated along with the training in every 5 epoch, in which the model takes `CIFAR-10` as ID data and `SVHN` as OOD data. As shown by the curves, a middle stage exists with better OOD performance than that of the final stage across different schedules. We empirically verify the existence of this phenomenon without schedule specificity. More explorations on other ID dataset and model structure reveal similar results in Figures 5, 6, and 7. The detailed information and discussion can refer to Appendix D.

Effects of the mask on mining atypical samples. Following our previous illustration of Figure 3(c), we scrutinize the change of training loss on a random batch of the training set in Figure 4(c). The results further explain why the loss value estimated by the UM can be used to force the model to forget atypical samples. It can be seen from Figure 4(c) that the loss is proportionally increased by randomly knocking out 2.5% weights. In this case, the estimated loss is more influenced by those who have a higher initial loss and are what we termed as atypical samples. By controlling the training loss to the estimated value, the model is encouraged to backtrack to a middle training stage where samples with high loss value have little influence on the forgetting process of the gradient ascent.

Exploration on revealing detection capability with model pruning. Although the large constrain on training loss can help reveal the model's OOD performance, the ID-ACC is undermined under such circumstances. Generally speaking, the proposed UM forces the model to forget the atypical samples and may result in lower test performance. To mitigate this issue, we further adopt pruning as a countermeasure to learn a mask instead of tuning the model parameters directly. In Figure 4(e), we experiment with various prune rates p and demonstrate that we can achieve the same or better OOD performance also by pruning. Specifically, our UMAP can achieve a lower FPR95 than pure pruning with the original objective. The prune rate can be selected from a wide range (e.g. $p \in [0.3, 0.9]$) to guarantee a fast convergence and effectiveness. Since pruning doesn't change the well-trained model parameters, it can preserve the performance of the original task. We also provide additional empirical results and corresponding discussion about the effectiveness of our UMAP in Appendix D.

5 Conclusion

In this work, we explore the intrinsic OOD detection capability of a well-trained model. Without involving any auxiliary outliers into training, we reveal the inconsistent trend between minimizing original training loss and gaining OOD detection capability. We further attribute it to the memorization behavior on atypical samples. To excavate the once-covered capability, we propose a new method, namely, Unleashing Mask (UM). Through UM, we construct model-level discrepancy that figures out the memorized atypical samples and utilizes the constrained gradient ascent to encourage forgetting. It better utilizes the given model for OOD detection via backtracking or sub-structure pruning. We hope our work could provide new insights for revisiting the model development in OOD detection.

REPRODUCIBILITY STATEMENT

For the experimental setups, we have provided the details in Section 4.1 and Appendixes A and C. We will also provide the anonymous repository about our source codes in the discussion phase for reviewing purposes to ensure the reproducibility of our experimental results.

REFERENCES

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron C. Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *ICML*, 2017.

Paul Barham, Aakanksha Chowdhery, Jeff Dean, Sanjay Ghemawat, Steven Hand, Daniel Hurt, Michael Isard, Hyeontaek Lim, Ruoming Pang, Sudip Roy, et al. Pathways: Asynchronous distributed dataflow for ml. In *MLSys*, 2022.

Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Atom: Robustifying out-of-distribution detection using outlier mining. In *ECML PKDD*, 2021.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *ICML*, 2006.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Xin Dong, Shangyu Chen, and Sinno Pan. Learning to prune deep neural networks via layer-wise optimal brain surgeon. In *NeurIPS*, 2017.

Xuefeng Du, Zhaoning Wang, Mu Cai, and Sharon Li. Towards unknown-aware learning with virtual outlier synthesis. In *ICLR*, 2022a.

Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. VOS: learning what you don't know by virtual outlier synthesis. In *ICLR*, 2022b.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *JMLR*, 2011.

Tom Fawcett. An introduction to roc analysis. In *Pattern Recognition Letters*, 2006.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. In *NeurIPS*, 2021.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*. MIT Press, 2016.

Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *ICLR*, 2016.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *Proceedings of International Conference on Learning Representations*, 2017a.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017b.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *ICML*, 2019a.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *ICLR*, 2019b.

Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. Deep anomaly detection with outlier exposure. In *ICLR*, 2019c.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *NeurIPS*, 2021.

Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. Do we need zero training loss after achieving zero training error? In *ICML*, 2020.

J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. In *Ann. Math. Statist.*, 1952.

Alex Krizhevsky. Learning multiple layers of features from tiny images. In *arXiv*, 2009.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *ICLR*, 2018a.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018b.

Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *ICLR*, 2018.

Ziqian Lin, Sreya Dutta Roy, and Yixuan Li. Mood: Multi-level out-of-distribution detection. In *CVPR*, 2021.

Weitang Liu, Xiaoyun Wang, John D Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *NeurIPS*, 2020.

Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *ICLR*, 2017.

Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.

Yifei Ming, Ying Fan, and Yixuan Li. Poem: Out-of-distribution detection with posterior sampling. In *ICML*, 2022.

Sina Mohseni, Mandar Pitale, J. B. S. Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. In *AAAI*, 2020.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NeurIPS Workshop*, 2011.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *CVPR*, 2015.

Vivek Ramanujan, Mitchell Wortsman, Aniruddha Kembhavi, Ali Farhadi, and Mohammad Rastegari. What's hidden in a randomly weighted neural network? In *CVPR*, 2020.

Mohammadreza Salehi, Hossein Mirzaei, Dan Hendrycks, Yixuan Li, Mohammad Hossein Rohban, and Mohammad Sabokrou. A unified survey on anomaly, novelty, open-set, and out-of-distribution detection: Solutions and future challenges. In *arXiv*, 2021.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. SSD: A unified framework for self-supervised outlier detection. In *ICLR*, 2021.

Jonathan Sorg, Richard L Lewis, and Satinder Singh. Reward design via online gradient ascent. In *NeurIPS*, 2010.

Yiyou Sun and Yixuan Li. Dice: Leveraging sparsification for out-of-distribution detection. In *ECCV*, 2022.

Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. In *NeurIPS*, 2021.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *ICML*, 2022.

Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. CSI: novelty detection via contrastive learning on distributionally shifted instances. In *NeurIPS*, 2020.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two sample. In *Biometrika*, 1933.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *CVPR*, 2018.

Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *ICML*, 2022.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. In *arXiv*, 2021.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. In *arXiv*, 2015.

Qing Yu and Kiyoharu Aizawa. Unsupervised out-of-distribution detection by maximum classifier discrepancy. In *ICCV*, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.

Bolei Zhou, Agata Lapedriza, Antonio Torralba, and Aude Oliva. Places: An image database for deep scene understanding. In *Journal of Vision*, 2017.

APPENDIX

REPRODUCIBILITY STATEMENT

We will provide the anonymous repository about our source codes in the discussion phase for reviewing purposes to ensure the reproducibility of our experimental results. Below we summarize some critical aspects to facilitate the reproducible results:

- **Datasets.** The datasets we used are all publicly accessible, which is introduced in Section 4.1. For methods involving auxiliary outliers, we strictly follow previous works (Sun et al., 2021; Du et al., 2022b) to avoid overlap between the auxiliary dataset (ImageNet-1k) (Deng et al., 2009) and any other OOD datasets.
- **Assumption.** We set our experiments to a post-hoc scenario where a well-trained model is available, and some parts of training samples are also available for subsequent fine-tuning.
- **Open source.** The code repository will be available in an anonymous repository for reviewing purposes. We provide a backbone of our experiments as well as several auxiliary components, such as score estimation.
- **Environment.** All experiments are conducted multiple runs on NVIDIA Tesla V100-SXM2-32GB GPUs with Python 3.6 and PyTorch 1.8.

A DETAILED INFORMATION ABOUT THE USED BASELINES

In this section, we provide the details about the baselines for the scoring functions and finetuning with auxiliary outliers, and the corresponding hyper-parameters that are considered in our work.

Maximum Softmax Probability (MSP). (Hendrycks & Gimpel, 2017a) proposes to use maximum softmax probability to discriminate ID and OOD samples. The score is defined as follows,

$$S_{\text{MSP}}(x; f) = \max_{c} P(y = c|x; f) = \max \ \texttt{softmax}(f(x)) \tag{4}$$

where f represents the given well-trained model and c is one of the classes $\mathcal{Y} = \{1, \dots, C\}$. The larger softmax score indicates the larger probability for a sample to be ID data, reflecting the model's confidence on the sample.

ODIN. (Liang et al., 2018) designed the ODIN score, leveraging the temperature scaling and tiny perturbations to widen the gap between the distributions of ID and OOD samples. The ODIN score is defined as follows,

$$S_{\text{ODIN}}(x; f) = \max_{c} P(y = c|\tilde{x}; f) = \max \ \texttt{softmax}(\frac{f(\tilde{x})}{T}) \tag{5}$$

where $\tilde{x}$ represents the perturbed samples (controled by ϵ), T represents the temperature. For fair comparison, we adopt the suggested hyperparameters (Liang et al., 2018): $\epsilon = 1.4 \times 10^{-3}$, $T = 1.0 \times 10^{4}$.

Mahalanobis. (Lee et al., 2018b) introduces a Mahalanobis distance-based confidence score, exploiting the feature space of the neural networks by inspecting the class conditional Gaussian distributions. The Mahalanobis distance score is defined as follows,

$$S_{\text{Mahalanobis}}(x; f) = \max_{c} -(f(x) - \hat{\mu}_c)^T \hat{\Sigma}^{-1} (f(x) - \hat{\mu}_c) \tag{6}$$

where $\hat{\mu}_c$ represents the estimated mean of multivariate Gaussian distribution of class c, $\hat{\Sigma}$ represents the estimated tied covariance of the C class-conditional Gaussian distributions.

Energy. (Liu et al., 2020) proposes to use the Energy of the predicted logits to distinguish the ID and OOD samples. The Energy score is defined as follows,

$$S_{\text{Energy}}(x; f) = -T \log \sum_{c=1}^{C} e^{f(x)_c / T} \tag{7}$$

where T represents the temperature parameter. As theoretically illustrated in Liu et al. (2020), a lower Energy score indicates a higher probability for a sample to be ID. Following (Liu et al., 2020), we fix the T to 1.0 throughout all experiments.

Outlier Exposure (OE). (Hendrycks et al., 2019b) initiates a promising approach towards OOD detections by involving outliers to force apart the distributions of ID and OOD samples. In the experiments, we use the cross-entropy from $f(x_{\text{out}})$to the uniform distribution as the $\mathcal{L}_{\text{OE}}$ (Lee et al., 2018a),

$$\mathcal{L}_f = \mathbb{E}_{D_{\text{in}}}\left[\ell_{\text{CE}}(f(x), y)\right] + \lambda \mathbb{E}_{D^{\text{s}}_{\text{out}}}\left[\log \sum_{c=1}^{C} e^{f(x)_c} - \mathbb{E}_{D^{\text{s}}_{\text{out}}}(f(x))\right] \tag{8}$$

Energy (w. D_{aux}). In addition to using the Energy as a post-hoc score to distinguish ID and OOD samples, (Liu et al., 2020) proposes an Energy-bounded objective to further separate the two distributions. The OE objective is as follows,

$$\mathcal{L}_{\text{OE}} = \mathbb{E}_{D^{\text{s}}_{\text{in}}}(\max(0, S_{\text{Energy}}(x, f) - m_{\text{in}}))^2 + \mathbb{E}_{D^{\text{s}}_{\text{out}}}(\max(0, m_{\text{out}} - S_{\text{Energy}}(x, f)))^2 \tag{9}$$

We keep the thresholds same to (Liu et al., 2020): $m_{\text{in}} = -25.0$, $m_{\text{out}} = -7.0$.

POEM. (Ming et al., 2022) explores the Thompson sampling strategy (Thompson, 1933) to make the most use of outliers to learn a tight decision boundary. Though given the POEM's nature to be orthogonal to other OE methods, we use the Energy(w. D_{aux}) as the backbone, which is the same as Eq.(9) in Liu et al. (2020). The details of Thompson sampling can refer to Ming et al. (2022).

B Detailed Realization of the Proposed Algorithms

In this section, we provide the detailed realization of our proposed Unleashing Mask (UM) (i.e., in Algorithm 1) and Unleashing Mask Adopt Pruning (UMAP) (i.e., in Algorithm 2).

To estimate the loss constrain ζ for forgetting, we need to randomly knock out parts of weights according to the given mask ratio δ. To be specific, we sample a score from a Gaussian distribution for every weight. Then we initialize a unit matrix for every layer of the model concerning the size of the layer. We formulate the mask m_δ according to the sampled scores. Find the threshold for each layer that is smaller than the score of the given mask ratio in that layer (termed as quantile). Then set all the ones, whose corresponding scores are more significant than the layers' thresholds, to zeros.

We dot-multiply every layer's weights with the formulated binary matrix as if we delete some parts of the weights. We input a batch of training samples to the masked model and treat the mean value of the outputs' cross-entropy loss as the loss constraint. After all of these have been done, we begin to fine-tune the model's weights with the loss constraint applied to the cross-entropy loss.

For UMAP, the only difference from UM is that, instead of fine-tuning the weights, we generate a popup score for every weight, and force the gradients to pass through the scores. In every iteration, we need to formulate a binary mask according to the given prune rate p. This is just what we do when estimating the loss constraint. For more details, it can refer to (Ramanujan et al., 2020). In Table 3, we summarize the overall comparison results of UM and UMAP to show their effectiveness.

Algorithm 1 Unleashing Mask (UM)

Input: well-trained model : θ, mask ratio: $\delta \in [0, 1]$, fine-tuning epochs of UM: k, training samples: $x \sim D^{\text{s}}_{\text{in}}$;
Output: fine-tuned model $\hat{\theta}$;

Initialize a popup score for every weight
for $w \in \theta$ **do**
  $s_w \sim N(\mu, \sigma^2)$
end for
Generate mask by the popup scores
for $l \in \theta_{\text{layers}}$ **do**
  $m^l_\delta = s^l >$ quantile(s^l, δ)
end for
Estimate loss constrain
$\zeta = \mathbb{E}_{x \sim D^{\text{s}}_{\text{in}}}(\hat{\mathcal{L}}_{\text{CE}}(x, m_\delta \odot \theta))$
Unleashing Mask: fine-tuning
for $t \in (1, \ldots, k)$ **do**
  $\theta^{(t+1)} = \theta^{(t)} - \eta \frac{\partial(|\mathcal{L}_{\text{CE}}(x,\theta) - \zeta| + \zeta)}{\partial \theta}$
end for

Algorithm 2 Unleashing Mask Adopt Pruning (UMAP)

Input: well-trained model : θ, mask ratio: $\delta \in [0, 1]$, fine-tuning epochs of UM: k, training samples: $x \sim D^{\text{s}}_{\text{in}}$, prune rate: p ;
Output: learnt binary mask $\hat{m}_p$;

Initialize a popup score for every weight
for $w \in \theta$ **do**
  $s_w \sim N(\mu, \sigma^2)$
end for
Generate mask by the popup scores
for $l \in \theta_{\text{layers}}$ **do**
  $m^l_\delta = s^l >$ quantile(s^l, δ)
end for
Estimate loss constrain
$\zeta = \mathbb{E}_{x \sim D^{\text{s}}_{\text{in}}}(\hat{\mathcal{L}}_{\text{CE}}(x, m_\delta \odot \theta))$
Unleashing Mask Adopt Pruning: fine-tuning
$s^{(1)} \sim N(\mu, \sigma^2)$
for $t \in (1, \ldots, k)$ **do**
  for $l \in \theta_{\text{layers}}$ **do**
    $m^l_p = s^{l(t)} >$ quantile$(s^{l(t)}, p)$
  end for
  $s^{(t+1)} = s^{(t)} - \eta \frac{\partial(|\mathcal{L}_{\text{CE}}(x, m_p \odot \theta) - \zeta| + \zeta)}{\partial \theta}$
end for

C ADDITIONAL SETUPS OF THE EXPERIMENTS

In this section, we describe more details about the experimental setups for our exploration.

Model setups. For DenseNet-101, we fix the growth rate and reduce rate to 12 and 0.5 respectively with the bottleneck block included in the backbone. We also explore the proposed UM on WideResNet (Zagoruyko & Komodakis, 2016) with 40 depth and 4 widen factor, which is termed as WRN-40-4. The batch size for both ID and OOD testing samples is 64, and the batch size of auxiliary samples is 2000. The λ in Eq.(8) is 0.5 to keep the OE loss comparable to the CE loss. As for the strategy of sampling outliers, we randomly retrieve 50000 samples from ImageNet-1k (Deng et al., 2009) for OE and Energy (w. D_{aux}) and 50000 samples using Thompson sampling (Ming et al., 2022) for POEM.

Learning rate schedules. We use 4 different learning rate schedules to demonstrate the existence of the once-covered OOD detection capability. For cosine annealing, we follow the common setups in Loshchilov & Hutter (2017); for linear schedule, the learning rate remains the same in the first one-third epochs, decreases linearly to the tenth of the initial rate in the middle one-third epochs, and decrease linearly to 1% of the initial rate in the last one-third epochs; for the multiple decay schedule, the learning rate decreases 10% of the initial rate (0.01) every 10% epochs (10 epochs); for the multiple step schedule, the learning rate decreases to 10% of the current rate every 30 epochs. All those learning rate schedules for our experiments are intuitively illustrated in Figure 4(a).

Table 3: Completed Results (%). Comparison with competitive OOD detection baselines. ↑ indicates higher values are better, and ↓ indicates lower values are better.

D_{in}	Method	AUROC↑	AUPR↑	FPR95↓	ID-ACC↑	w./w.o D_{aux}
CIFAR-10	MSP(Hendrycks & Gimpel, 2017a)	89.90 ± 0.30	91.48 ± 0.43	60.08 ± 0.76	94.01 ± 0.08	
	ODIN(Liang et al., 2018)	91.46 ± 0.56	91.67 ± 0.58	42.31 ± 1.38	94.01 ± 0.08	
	Mahalanobis(Lee et al., 2018b)	75.10 ± 1.04	72.32 ± 1.92	61.35 ± 1.25	94.01 ± 0.08	
	Energy(Liu et al., 2020)	92.07 ± 0.22	92.72 ± 0.39	42.69 ± 1.31	94.01 ± 0.08	
	Energy+UM (ours)	93.73 ± 0.36	94.27 ± 0.60	33.29 ± 1.70	92.80 ± 0.47	
	Energy+UMAP (ours)	$\mathbf{93.97 \pm 0.11}$	$\mathbf{94.38 \pm 0.06}$	$\mathbf{30.71 \pm 1.94}$	$\mathbf{94.01 \pm 0.08}$	
	OE(Hendrycks et al., 2019b)	97.07 ± 0.01	97.31 ± 0.05	13.80 ± 0.28	92.59 ± 0.32	✓
	Energy (w. D_{aux})(Liu et al., 2020)	94.58 ± 0.64	94.69 ± 0.65	18.79 ± 2.31	80.91 ± 3.13	✓
	POEM(Ming et al., 2022)	94.37 ± 0.07	94.51 ± 0.06	18.50 ± 0.33	77.24 ± 2.22	✓
	OE+UM (ours)	$\mathbf{97.60 \pm 0.03}$	$\mathbf{97.87 \pm 0.02}$	$\mathbf{11.22 \pm 0.16}$	$\mathbf{93.66 \pm 0.12}$	✓
	Energy+UM (ours)	93.02 ± 0.42	92.36 ± 0.38	24.41 ± 1.65	71.97 ± 0.92	✓
	POEM+UM (ours)	93.04 ± 0.02	92.99 ± 0.02	23.52 ± 0.16	67.41 ± 0.27	✓
	OE+UMAP (ours)	97.48 ± 0.01	97.74 ± 0.00	12.21 ± 0.09	93.44 ± 0.21	✓
	Energy+UMAP (ours)	95.63 ± 1.15	95.92 ± 1.17	17.51 ± 2.59	88.12 ± 4.22	✓
	POEM+UMAP (ours)	94.18 ± 2.98	94.15 ± 3.46	20.55 ± 8.70	76.62 ± 17.95	✓
CIFAR-100	MSP(Hendrycks & Gimpel, 2017a)	74.06 ± 0.69	75.37 ± 0.73	83.14 ± 0.87	74.86 ± 0.21	
	ODIN(Liang et al., 2018)	76.18 ± 0.14	76.49 ± 0.20	78.93 ± 0.31	74.86 ± 0.21	
	Mahalanobis(Lee et al., 2018b)	63.90 ± 1.91	64.31 ± 0.91	78.79 ± 0.50	74.86 ± 0.21	
	Energy(Liu et al., 2020)	$\mathbf{76.29 \pm 0.24}$	$\mathbf{77.06 \pm 0.55}$	78.46 ± 0.06	74.86 ± 0.21	
	Energy+UM (ours)	76.22 ± 0.42	76.39 ± 1.03	74.05 ± 0.55	64.55 ± 0.24	
	Energy+UMAP (ours)	75.57 ± 0.59	75.66 ± 0.07	$\mathbf{72.21 \pm 1.46}$	$\mathbf{74.86 \pm 0.21}$	
	OE(Hendrycks et al., 2019b)	90.55 ± 0.87	90.34 ± 0.94	34.73 ± 3.85	73.59 ± 0.30	✓
	Energy (w. D_{aux})(Liu et al., 2020)	88.92 ± 0.57	89.13 ± 0.56	37.90 ± 2.59	57.85 ± 2.65	✓
	POEM(Ming et al., 2022)	88.95 ± 0.54	88.94 ± 0.31	38.10 ± 1.30	56.18 ± 1.92	✓
	OE+UM (ours)	91.04 ± 0.11	91.13 ± 0.24	34.71 ± 0.81	$\mathbf{75.15 \pm 0.18}$	✓
	Energy+UM (ours)	90.39 ± 0.40	90.14 ± 0.45	32.65 ± 3.13	71.95 ± 0.23	✓
	POEM+UM (ours)	$\mathbf{91.18 \pm 0.35}$	$\mathbf{91.45 \pm 0.27}$	$\mathbf{30.78 \pm 1.76}$	70.17 ± 0.01	✓
	OE+UMAP (ours)	91.10 ± 0.16	90.99 ± 0.23	33.62 ± 0.26	74.76 ± 0.11	✓
	Energy+UMAP (ours)	90.52 ± 0.26	90.46 ± 0.50	32.17 ± 0.30	72.76 ± 0.18	✓
	POEM+UMAP (ours)	91.10 ± 0.29	91.41 ± 0.28	31.02 ± 1.70	71.05 ± 0.04	✓

D ADDITIONAL EXPERIMENT RESULTS AND ABLATION STUDY

In this section, we provide more experiment results. We first show the fine-grained results on CIFAR-10 and CIFAR-100, then conduct the experiments under a different model structure (i.e., WRN-40-4), and finally apply an additional ablation study on the proposed UM and UMAP. The mean and variance of all metrics (ID-ACC, AUROC, AUPR, FPR95) are reported based on multiple independent trials.

Table 4: Fine-grained Results (%) of DenseNet-101 on CIFAR-10. Comparison on different OOD benchmark datasets respectively. ↑ indicates higher values are better, and ↓ indicates lower values are better.

ID dataset	Method	OOD dataset CIFAR-100 FPR95↓	CIFAR-100 AUROC↑	Textures FPR95↓	Textures AUROC↑	Places365 FPR95↓	Places365 AUROC↑
CIFAR-10	**Energy + UMAP**	54.95 ± 2.61	87.72 ± 1.05	33.59 ± 1.32	92.67 ± 0.23	32.80 ± 4.14	93.57 ± 1.12
	OE	59.29 ± 1.30	88.51 ± 0.22	2.89 ± 0.30	99.16 ± 0.05	11.14 ± 1.11	97.50 ± 0.19
	Energy (w. D_{aux})	79.88 ± 2.47	74.99 ± 2.40	4.27 ± 0.57	98.80 ± 0.19	14.22 ± 3.99	97.07 ± 0.72
	POEM	82.30 ± 1.57	72.74 ± 1.42	1.91 ± 0.41	99.40 ± 0.10	11.24 ± 2.70	96.67 ± 0.48
	OE + UM (ours)	$\mathbf{55.74 \pm 1.47}$	$\mathbf{89.53 \pm 0.18}$	$\mathbf{1.42 \pm 0.15}$	$\mathbf{99.49 \pm 0.04}$	$\mathbf{7.77 \pm 0.69}$	$\mathbf{98.15 \pm 0.08}$
	Energy (w. D_{aux})+ UM (ours)	84.52 ± 0.01	70.09 ± 0.47	8.30 ± 0.88	97.76 ± 0.05	20.27 ± 1.30	96.06 ± 0.31
	POEM + UM (ours)	84.87 ± 1.56	68.97 ± 0.39	4.73 ± 0.52	98.88 ± 0.13	19.83 ± 0.34	96.35 ± 0.09
	OE + UMAP (ours)	59.05 ± 1.41	89.14 ± 0.14	1.86 ± 0.07	99.35 ± 0.00	8.21 ± 0.12	98.07 ± 0.03
	Energy (w. D_{aux}) + UMAP (ours)	75.18 ± 4.96	80.93 ± 4.49	2.24 ± 1.34	99.25 ± 0.29	9.30 ± 2.12	97.90 ± 0.40
	POEM + UMAP (ours)	79.33 ± 4.14	76.89 ± 4.86	2.10 ± 1.37	99.34 ± 0.30	9.94 ± 6.92	98.01 ± 1.11
	Method	**SUN** FPR95↓	**SUN** AUROC↑	**LSUN** FPR95↓	**LSUN** AUROC↑	**iNaturalist** FPR95↓	**iNaturalist** AUROC↑
	Energy + UMAP	29.05 ± 2.78	94.41 ± 0.73	2.31 ± 0.88	99.42 ± 0.04	47.22 ± 14.03	92.63 ± 2.14
	OE	8.38 ± 0.71	98.00 ± 0.14	5.90 ± 1.43	98.60 ± 0.21	5.09 ± 0.64	98.76 ± 0.11
	Energy (w. D_{aux})	10.30 ± 3.82	97.77 ± 0.64	12.80 ± 4.67	96.08 ± 1.38	6.93 ± 1.86	98.40 ± 0.32
	POEM	8.39 ± 2.42	98.16 ± 0.43	9.69 ± 1.89	97.25 ± 0.58	3.78 ± 0.90	98.99 ± 0.17
	OE + UM (ours)	$\mathbf{5.51 \pm 0.44}$	$\mathbf{98.55 \pm 0.07}$	$\mathbf{3.51 \pm 0.43}$	$\mathbf{98.93 \pm 0.09}$	$\mathbf{2.87 \pm 0.49}$	$\mathbf{99.14 \pm 0.09}$
	Energy (w. D_{aux})+ UM (ours)	16.13 ± 1.86	96.84 ± 0.30	23.27 ± 2.40	92.11 ± 0.94	11.20 ± 2.35	97.54 ± 0.42
	POEM + UM (ours)	16.16 ± 0.57	97.01 ± 0.08	25.69 ± 0.15	93.38 ± 0.27	9.30 ± 1.60	98.05 ± 0.24
	OE + UMAP (ours)	6.16 ± 0.02	98.49 ± 0.01	4.53 ± 0.16	98.86 ± 0.06	3.40 ± 0.74	98.96 ± 0.09
	Energy (w. D_{aux}) + UMAP (ours)	6.67 ± 1.50	98.40 ± 0.32	23.50 ± 5.61	94.78 ± 2.04	3.77 ± 2.14	98.93 ± 0.46
	POEM + UMAP (ours)	7.00 ± 5.80	98.46 ± 0.96	21.17 ± 12.84	94.74 ± 3.67	3.63 ± 2.78	98.99 ± 0.54

Fine-grained results on OOD data. In order to further figure out the effectiveness of the proposed UM and UMAP on different OOD datasets, we further report the fine-grained results of our experiments on CIFAR-10 and CIFAR-100 with 6 OOD datasets (CIFAR-10/CIFAR-100, textures, Places365, SUN, LSUN, iNaturalist). The results on the 6 OOD datasets show the general effectiveness of the proposed UM as well as UMAP. In Table 4, **OE + UM** can outperform all the OOD baselines, and further improve the OOD performance even though the original detection performance is already well. By equipping with our proposed UM and UMAP, the baselines can outperform their counterparts on most of the OOD datasets. For instance, the FPR95 can decrease from 1.91 to 1.42. In Table 5, we also take a closer check about results on CIFAR-100 with 6 OOD datasets. Our proposed method can almost improve all competitive baselines (either the scoring functions or the finetuning with auxiliary outliers) on the 6 OOD datasets. In both w. D_{aux} and w.o. D_{aux} scenarios, Unleashing Mask can significantly excavate the intrinsic OOD detection capability of the model. In addition to unleashing the excellent OOD performance, UMAP can also maintain the high ID-ACC by learning a binary mask instead of tuning the well-trained original parameters directly.

Experiment on different model structure. Following 4.2, we additionally conduct critical experiments on the WRN-40-4 (Lin et al., 2021) backbone to demonstrate the effectiveness of the proposed UM and UMAP. In Figure 7, we can find during the model training phase on ID data, there also exists the once-covered OOD detection capability can be explored in later development. In Table 6, we show the comparison of multiple OOD detection baselines, evaluating the OOD performance on the 7 OOD datasets mentioned in Section 4.1. The results again demonstrate that our proposed method indeed excavates the intrinsic detection capability and improves the performance.

As for the fine-grained results of WRN-40-4, we report results on 6 OOD datasets respectively. When trained on CIFAR-10, UM can outstrip all the scoring function baselines on 5 OOD datasets except Textures on which Mahalanobis performs better while UMAP still has excellent OOD performance ranking only second to UM. When trained on CIFAR-100, UM and UMAP can also outperform the baselines on most OOD datasets. The fine-grained results of WRN-40-4 further demonstrate the effectiveness of the proposed UM/UMAP on other architectures.

Additional results about the general existence of once-covered OOD detection capability. In Section 4.3, we display the once-covered OOD detection capability on CIFAR-10 using SVHN as the OOD dataset. Here, we additionally verify the previously observed trend during training when training DenseNet-101 on CIFAR-100 using iNaturalist as an OOD dataset. In Figure 5, we trace the three evaluation metrics during training on CIFAR-100 using 4 different learning rate schedules. Consistent with the original experiment, we still use iNaturalist as the OOD dataset. It can be seen for all the three metrics that exists a middle stage where the model has the better OOD detection

Table 5: Fine-grained Results (%) of DenseNet-101 on CIFAR-100. Comparison on different OOD benchmark datasets respectively. ↑ indicates higher values are better, and ↓ indicates lower values are better.

ID dataset	Method	OOD dataset CIFAR-10 FPR95↓	CIFAR-10 AUROC↑	Textures FPR95↓	Textures AUROC↑	Places365 FPR95↓	Places365 AUROC↑
CIFAR-100	MSP	83.53 ± 0.33	75.11 ± 0.27	86.90 ± 0.18	71.45 ± 0.40	85.83 ± 0.48	70.54 ± 0.42
	ODIN	85.29 ± 0.17	73.31 ± 0.24	86.45 ± 1.27	71.91 ± 0.27	84.35 ± 0.64	73.58 ± 0.51
	Mahalanobis	98.25 ± 0.05	49.60 ± 1.51	$\mathbf{33.06 \pm 3.76}$	$\mathbf{90.19 \pm 1.21}$	95.20 ± 0.49	53.69 ± 1.55
	Energy	$\mathbf{82.16 \pm 0.59}$	$\mathbf{75.31 \pm 0.21}$	90.20 ± 0.30	68.98 ± 0.34	82.39 ± 0.97	73.78 ± 0.66
	Energy+UM (ours)	89.62 ± 0.07	66.12 ± 0.93	86.99 ± 1.22	65.39 ± 1.44	$\mathbf{77.30 \pm 2.08}$	$\mathbf{76.06 \pm 1.05}$
	OE	90.97 ± 0.46	69.02 ± 0.39	14.36 ± 0.25	95.92 ± 0.12	40.19 ± 6.97	90.70 ± 2.01
	Energy (w. D_{aux})	96.14 ± 0.06	57.52 ± 0.88	9.02 ± 0.06	97.16 ± 0.26	35.18 ± 4.73	93.29 ± 1.14
	POEM	96.19 ± 0.16	55.82 ± 1.05	7.63 ± 1.40	97.69 ± 0.06	32.67 ± 3.73	93.94 ± 0.68
	OE + UM (ours)	89.61 ± 0.08	$\mathbf{71.24 \pm 0.08}$	16.78 ± 0.25	95.60 ± 0.08	39.77 ± 0.34	91.07 ± 0.13
	Energy (w. D_{aux})+ UM (ours)	95.38 ± 0.45	63.41 ± 0.14	6.41 ± 0.83	97.77 ± 0.33	30.96 ± 3.61	92.85 ± 0.69
	POEM + UM (ours)	95.78 ± 0.14	60.23 ± 0.70	$\mathbf{5.17 \pm 0.18}$	$\mathbf{98.53 \pm 0.03}$	$\mathbf{23.90 \pm 0.84}$	$\mathbf{95.45 \pm 0.11}$
	OE + UMAP (ours)	$\mathbf{90.72 \pm 0.35}$	69.76 ± 0.25	15.32 ± 0.23	95.72 ± 0.01	36.42 ± 1.91	92.08 ± 0.49
	Energy (w. D_{aux}) + UMAP (ours)	95.39 ± 0.10	63.26 ± 0.18	6.52 ± 0.44	97.83 ± 0.18	31.18 ± 0.43	93.13 ± 0.41
	POEM + UMAP (ours)	95.69 ± 0.17	61.62 ± 0.24	5.23 ± 0.58	98.52 ± 0.01	26.06 ± 1.16	94.91 ± 0.25
	Method	**SUN** FPR95↓	**SUN** AUROC↑	**LSUN** FPR95↓	**LSUN** AUROC↑	**iNaturalist** FPR95↓	**iNaturalist** AUROC↑
	MSP	88.75 ± 0.23	66.75 ± 0.25	67.83 ± 1.37	82.94 ± 0.32	85.00 ± 0.73	76.62 ± 0.25
	ODIN	88.49 ± 0.99	69.64 ± 0.61	34.80 ± 2.55	93.92 ± 0.75	81.67 ± 2.77	78.36 ± 1.57
	Mahalanobis	95.53 ± 0.37	54.37 ± 1.35	89.31 ± 4.83	43.19 ± 16.36	93.63 ± 1.19	49.60 ± 1.51
	Energy	97.17 ± 0.92	69.04 ± 0.83	35.09 ± 3.17	93.49 ± 0.87	85.70 ± 2.14	75.82 ± 1.72
	Energy+UM (ours)	$\mathbf{81.96 \pm 2.26}$	$\mathbf{71.47 \pm 1.88}$	$\mathbf{22.54 \pm 5.93}$	$\mathbf{94.98 \pm 1.75}$	$\mathbf{74.28 \pm 3.72}$	$\mathbf{80.72 \pm 3.75}$
	OE	44.47 ± 9.10	90.70 ± 2.01	$\mathbf{5.75 \pm 1.18}$	98.57 ± 0.13	25.51 ± 4.12	94.46 ± 0.88
	Energy (w. D_{aux})	32.69 ± 5.69	93.63 ± 1.48	55.75 ± 4.31	87.96 ± 1.03	17.34 ± 4.54	96.50 ± 0.81
	POEM	30.45 ± 5.11	94.26 ± 0.90	46.68 ± 3.59	90.30 ± 2.17	16.50 ± 2.09	96.63 ± 0.23
	OE + UM (ours)	44.23 ± 0.20	90.28 ± 0.03	5.80 ± 0.33	98.63 ± 0.03	26.72 ± 1.95	94.51 ± 0.43
	Energy (w. D_{aux})+ UM (ours)	28.98 ± 3.15	93.18 ± 0.69	37.56 ± 4.81	91.98 ± 0.83	10.83 ± 2.06	97.09 ± 0.69
	POEM + UM (ours)	$\mathbf{21.34 \pm 1.07}$	$\mathbf{95.76 \pm 0.22}$	33.74 ± 6.22	94.43 ± 1.18	$\mathbf{8.85 \pm 0.23}$	$\mathbf{97.93 \pm 0.03}$
	OE + UMAP (ours)	39.58 ± 2.02	91.37 ± 0.52	5.77 ± 0.71	$\mathbf{98.64 \pm 0.12}$	23.33 ± 1.24	95.08 ± 0.22
	Energy (w. D_{aux}) + UMAP (ours)	29.65 ± 1.06	93.38 ± 0.19	35.94 ± 0.75	92.08 ± 0.39	13.96 ± 2.48	96.87 ± 0.15
	POEM + UMAP (ours)	23.73 ± 0.71	95.25 ± 0.06	33.09 ± 5.94	93.57 ± 0.88	9.76 ± 1.09	97.77 ± 0.24

Table 6: Results (%) of WRN-40-4. Comparison with competitive OOD detection baselines. We respectively train WRN-40-4 on CIFAR-10 and CIFAR-100. For those methods involving outliers, we retrieve 5000 samples from ImageNet-1k. ↑ indicates higher values are better, and ↓ indicates lower values are better.

D_{in}	Method	AUROC↑	AUPR↑	FPR95↓	ID-ACC↑
CIFAR-10	MSP(Hendrycks & Gimpel, 2017a)	87.12 ± 0.25	87.84 ± 0.30	68.29 ± 0.96	93.86 ± 0.19
	ODIN(Liang et al., 2018)	83.29 ± 0.72	82.74 ± 0.79	65.68 ± 0.77	93.86 ± 0.19
	Mahalanobis(Lee et al., 2018b)	77.57 ± 0.28	76.11 ± 0.10	61.18 ± 0.10	93.86 ± 0.19
	Energy(Liu et al., 2020)	87.69 ± 0.54	88.16 ± 0.69	58.47 ± 1.94	93.86 ± 0.19
	Energy+UM (ours)	$\mathbf{91.74 \pm 0.43}$	$\mathbf{92.67 \pm 0.52}$	$\mathbf{40.40 \pm 1.32}$	92.68 ± 0.23
	Energy+UMAP (ours)	88.84 ± 1.02	89.31 ± 1.44	50.23 ± 2.25	$\mathbf{93.86 \pm 0.19}$
CIFAR-100	MSP(Hendrycks & Gimpel, 2017a)	72.34 ± 0.63	72.69 ± 0.44	85.40 ± 0.59	75.01 ± 0.07
	ODIN(Liang et al., 2018)	68.78 ± 0.67	66.92 ± 0.72	85.28 ± 0.64	75.01 ± 0.07
	Mahalanobis(Lee et al., 2018b)	68.20 ± 0.99	68.30 ± 1.15	76.46 ± 2.02	75.01 ± 0.07
	Energy(Liu et al., 2020)	74.00 ± 0.41	73.02 ± 0.47	81.37 ± 0.08	75.01 ± 0.07
	Energy+UM (ours)	76.07 ± 0.04	76.94 ± 0.06	74.29 ± 1.66	59.08 ± 2.75
	Energy+UMAP (ours)	$\mathbf{77.35 \pm 0.78}$	$\mathbf{77.43 \pm 0.91}$	$\mathbf{68.20 \pm 0.06}$	$\mathbf{75.01 \pm 0.07}$

capability (For FPR95, it is smaller (better) in the middle stage; for AUROC and AUPR, they are higher (better) in the middle stage). Besides that, we also look into the change of OOD performance on other architecture (e.g., WRN-40-4) in Figure 6 and Figure 7. In Figure 6, we display the curves of three metrics of WRN-40-4 when trained on CIFAR-10 with SVHN and Textures as OOD datasets. The trend that the OOD performance first goes better and then converges to worse OOD performance can be reflected. In Figure 7, we continually provide curves of the three metrics of WRN-40-4 during training on CIFAR-100 with iNaturalist, Places365, and SUN as OOD datasets. A clear better middle stage can still be excavated in this scenario.

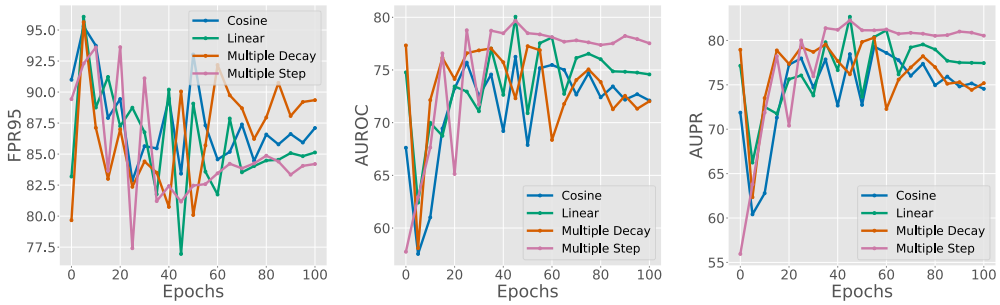
UMAP: adopting pruning on UM. We conduct various experiments to see whether pruning has an impact on Unleashing Mask itself. To be specific, we expect the pruning to learn a mask on the given model while not impairing the excellent OOD performance that UM brings. In Figure 8, it presents that pruning from a wide range (e.g. $p \in [0.3, 0.9]$) can well maintain the effectiveness of UM while possessing a terrific convergence trend. For simplicity, we use prune to indicate the

Table 7: Fine-grained Results (%) of WRN-40-4 on CIFAR-10. Comparison on different OOD benchmark datasets. ↑ indicates higher values are better, and ↓ indicates lower values are better.

ID dataset	Method	OOD dataset					
		CIFAR-100		Textures		Places365	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-10	MSP	70.96 ± 0.70	86.08 ± 0.08	68.81 ± 1.29	86.53 ± 0.83	68.31 ± 0.25	86.71 ± 0.13
	ODIN	64.97 ± 0.08	83.36 ± 0.11	66.86 ± 2.24	81.34 ± 0.81	66.49 ± 1.16	83.47 ± 0.93
	Mahalanobis	79.84 ± 0.55	70.33 ± 0.24	$\mathbf{22.56 \pm 0.08}$	$\mathbf{94.07 \pm 0.04}$	85.09 ± 0.59	67.90 ± 0.37
	Energy	61.09 ± 0.58	86.66 ± 0.04	64.29 ± 1.72	85.56 ± 0.53	55.32 ± 0.13	88.29 ± 0.26
	Energy+UM (ours)	$\mathbf{57.21 \pm 1.41}$	$\mathbf{87.56 \pm 0.15}$	46.49 ± 1.03	89.74 ± 0.45	$\mathbf{40.68 \pm 4.46}$	$\mathbf{92.51 \pm 0.97}$
	Energy+UMAP (ours)	65.45 ± 1.10	84.65 ± 0.95	59.14 ± 1.64	85.27 ± 1.74	48.16 ± 1.89	90.43 ± 0.47
	Method	**SUN**		**LSUN**		**iNaturalist**	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
	MSP	68.62 ± 0.50	86.95 ± 0.23	52.97 ± 3.07	92.41 ± 0.18	76.05 ± 0.01	83.44 ± 0.36
	ODIN	65.47 ± 0.78	83.79 ± 1.10	31.89 ± 3.44	94.34 ± 0.89	79.28 ± 0.18	79.80 ± 0.35
	Mahalanobis	82.92 ± 0.28	70.52 ± 0.47	64.31 ± 0.57	67.75 ± 0.55	81.50 ± 2.91	74.97 ± 2.91
	Energy	54.88 ± 0.18	88.67 ± 0.30	24.99 ± 1.38	95.98 ± 0.37	75.89 ± 0.85	82.40 ± 0.22
	Energy+UM (ours)	$\mathbf{38.92 \pm 3.46}$	$\mathbf{92.98 \pm 0.95}$	$\mathbf{8.38 \pm 0.77}$	$\mathbf{98.18 \pm 0.16}$	$\mathbf{66.02 \pm 6.70}$	$\mathbf{85.22 \pm 3.42}$
	Energy+UMAP (ours)	45.94 ± 2.64	91.27 ± 0.56	14.10 ± 0.04	97.46 ± 0.14	74.69 ± 0.15	81.13 ± 1.12

Table 8: Fine-grained Results (%) of WRN-40-4 on CIFAR-100. Comparison on different OOD benchmark datasets. ↑ indicates higher values are better, and ↓ indicates lower values are better.

ID dataset	Method	OOD dataset					
		CIFAR-10		Textures		Places365	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
CIFAR-100	MSP	83.83 ± 0.29	75.50 ± 0.21	86.15 ± 0.23	72.36 ± 0.40	86.72 ± 0.29	69.60 ± 0.27
	ODIN	83.70 ± 0.30	74.32 ± 0.03	81.57 ± 1.74	71.67 ± 0.28	88.07 ± 0.11	64.83 ± 1.36
	Mahalanobis	96.89 ± 0.11	68.78 ± 0.67	$\mathbf{31.02 \pm 2.04}$	$\mathbf{91.85 \pm 0.91}$	93.34 ± 0.44	61.28 ± 1.62
	Energy	$\mathbf{81.32 \pm 0.47}$	$\mathbf{77.49 \pm 0.26}$	86.38 ± 0.49	73.50 ± 0.45	84.45 ± 0.38	69.82 ± 0.63
	Energy+UM (ours)	89.23 ± 1.51	63.85 ± 1.73	78.90 ± 0.07	72.58 ± 1.33	$\mathbf{80.46 \pm 1.99}$	70.49 ± 1.01
	Energy+UMAP (ours)	94.11 ± 0.72	60.77 ± 0.96	66.94 ± 4.49	75.82 ± 5.33	82.59 ± 0.27	$\mathbf{71.92 \pm 3.58}$
	Method	**SUN**		**LSUN**		**iNaturalist**	
		FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
	MSP	88.88 ± 0.83	65.22 ± 0.85	78.56 ± 0.66	79.10 ± 0.53	86.72 ± 0.29	$73,75 \pm 0.56$
	ODIN	91.00 ± 0.10	59.06 ± 1.81	70.14 ± 2.42	84.03 ± 0.86	87.86 ± 1.13	64.52 ± 1.51
	Mahalanobis	94.22 ± 0.01	60.09 ± 1.56	89.73 ± 2.87	40.81 ± 3.07	87.25 ± 3.28	74.98 ± 2.85
	Energy	88.35 ± 0.52	64.04 ± 0.76	59.84 ± 0.06	87.91 ± 0.53	88.91 ± 0.78	67.81 ± 0.91
	Energy+UM (ours)	84.04 ± 0.09	67.19 ± 0.14	33.87 ± 1.21	92.29 ± 0.02	76.91 ± 6.07	79.28 ± 4.17
	Energy+UMAP (ours)	$\mathbf{80.53 \pm 1.31}$	$\mathbf{72.68 \pm 4.50}$	$\mathbf{27.79 \pm 2.19}$	$\mathbf{93.39 \pm 0.57}$	$\mathbf{55.53 \pm 4.61}$	$\mathbf{85.65 \pm 0.83}$



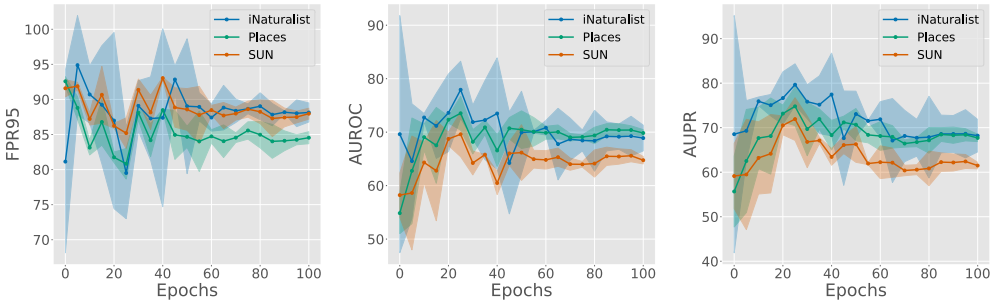
(a) FPR95 Curves (b) AUROC Curves (c) AUPR Curves

Figure 5: Ablation studies on three metrics with 4 different learning rate schedules. The model is DenseNet-101 trained on CIFAR-100 with iNaturalist as the OOD dataset. (a) change of FPR95 throughout the pruning phase when training on CIFAR-100; (b) change of AUROC throughout the pruning phase when training on CIFAR-100; (c) change of AUPR throughout the pruning phase when training on CIFAR-100. It demonstrates a better middle stage exists according to the three metrics.

original pruning approach and UMAP indicate UM with pruning on the mask with our newly designed forgetting objective in Figure 8. In Figure 8(a), the solid lines represent the proposed UMAP and the dashed lines represent only pruning the well-trained model at prune rates 0.2, 0.5, and 0.8. While the model's OOD performance can't be improved (not better than the baseline) through only pruning,

(a) FPR95 Curves (b) AUROC Curves (c) AUPR Curves

Figure 6: Ablation studies on three metrics of WRN-40-4 with CIFAR-10 as ID dataset, SVHN, and Textures as OOD datasets. (a) change of FPR95 throughout the pruning phase when training on CIFAR-10; (b) change of AUROC throughout the pruning phase when training on CIFAR-10; (c) change of AUPR throughout the pruning phase when training on CIFAR-10. It demonstrates a better middle stage exists according to the three metrics.



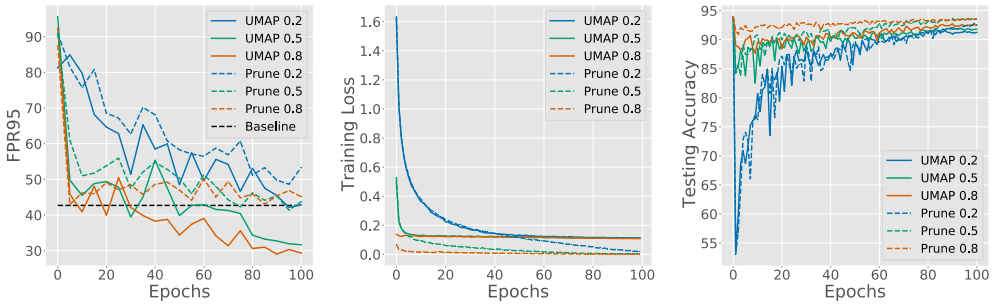
(a) FPR95 Curves (b) AUROC Curves (c) AUPR Curves

Figure 7: Ablation studies on three metrics of WRN-40-4 with CIFAR-100 as ID dataset, iNaturalist, Places365, and SUN as OOD datasets. (a) change of FPR95 throughout the pruning phase when training on CIFAR-100; (b) change of AUROC throughout the pruning phase when training on CIFAR-100; (c) change of AUPR throughout the pruning phase when training on CIFAR-100. It demonstrates a better middle stage exists according to the three metrics.

using our proposed forgetting objective for the loss constrain can significantly bring out better OOD performance at a wide range of mask rates (e.g. $p \in [0.5, 0.8]$). In Figure 8(b), we intuitively reflect the effect of the estimated loss constraint by the initialized mask which redirects the gradients when the loss reaches the value, while the loss will just approach 0 when pruning only. In Figure 8(c), we can see that ID-ACC for both UMAP and Prune can converge to approximately the same high level ($92 \sim 94$), though we can simply remove the learned mask to recover the original ID-ACC.

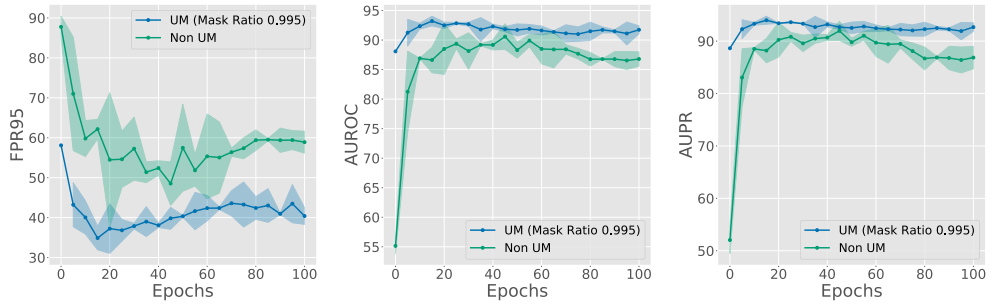
The effectiveness of UM. In Figure 9, we present the FPR95, AUROC, and AUPR curves during training to show the comparison of the original training and our proposed UM on ID data. We observe that training using UM can consistently outperform than the vanilla model training, either for the final stage or the middle stage with the best OOD detection performance indicated by the FPR95 curve. In Figure 10, we also adopt different mask rates for the initialized loss estimation for forgetting the atypical samples. In a range of the mask rate (e.g. $\delta \in [0.95, 0.99]$), we can see the proposed UM can excavate the intrinsic OOD detection capability. Due to the general effectiveness of UM, the choice of mask rate is relatively easy for UM's practicality.

Fine-grained comparison of the model weights. We display the weights of the original model, pruned model, and the UMAP model respectively in Figure 11. The histograms show that the



(a) FPR95 Curves (b) Train Loss Curves (c) Test Acc Curves

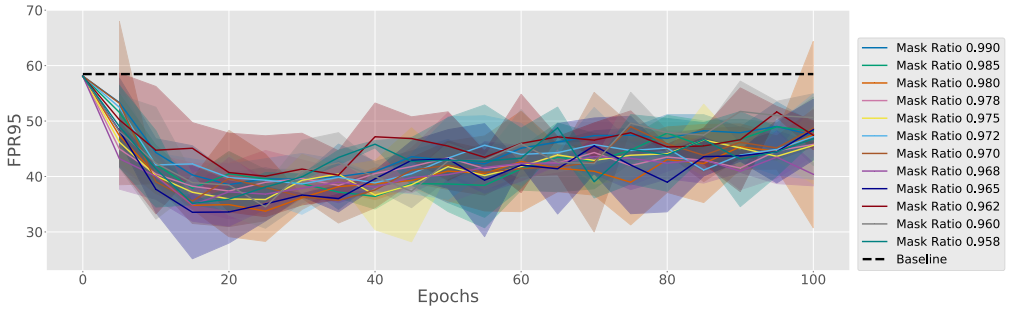
Figure 8: Ablation studies on Prune Rate of UMAP. (a) change of OOD performance throughout the pruning phase; (b) training loss converges to estimated loss constraint properly; (c) though ID-ACC is not taken into consideration for UMAP, it still raise high after training for 100 epochs.



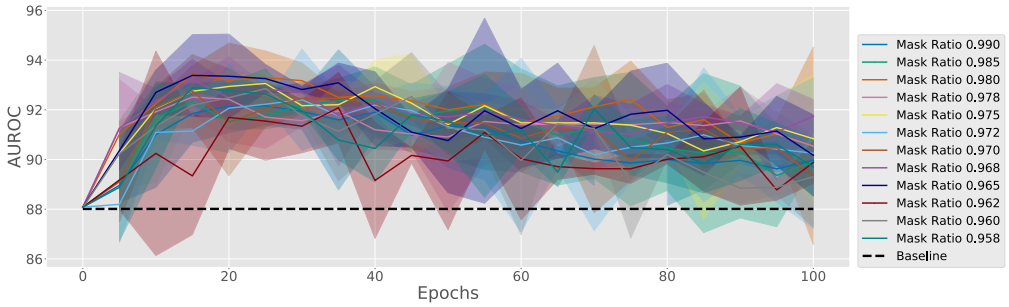
(a) FPR95 Curves (b) AUROC Curves (c) AUPR Curves

Figure 9: Ablation studies to reflect the effectiveness of UM. The mask ratio of UM is 99.5%. (a) change of FPR95 throughout the training phase on CIFAR-10; (b) change of AUROC throughout the training phase on CIFAR-10; (c) change of AUPR throughout the training phase on CIFAR-10.

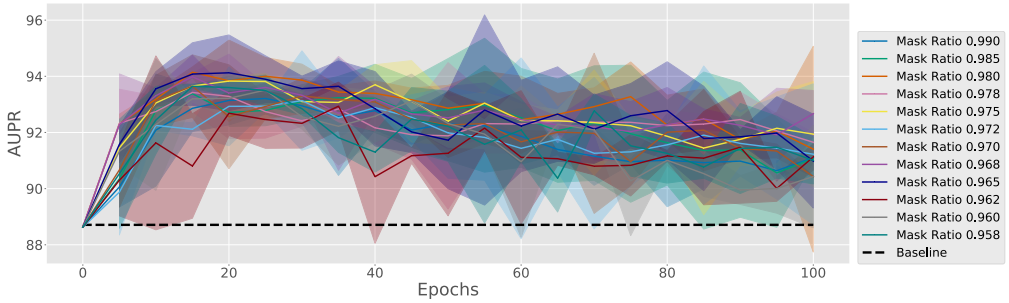
adopted pruning algorithm tends to choose weights far from 0 for the first convolution layer, shown in Figure 11(a). However, for almost all layers (from the 2nd to the 98th), the pruning chooses weights with no respect to the value of weights, shown in Figure 11(b). For the "head" of the model (the fully connected layer), the pruning algorithm itself still keeps its behavior on the first layer, while UMAP forces the prune algorithm to choose weights near 0, shown in Figure 11(c), indicating that forgetting learned atypical samples doesn't necessarily correspond to larger weights or smaller weights.



(a) FPR95 Curves

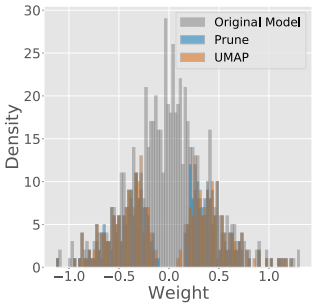


(b) AUROC Curves

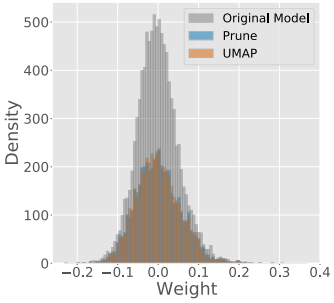


(c) AUPR Curves

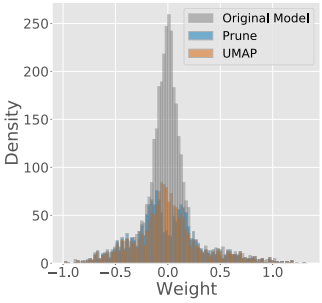
Figure 10: Ablation studies of WRN-40-4 on various Mask Ratios. The mask rate is from 95.8% to 99.0%. (a) change of FPR95 throughout the training on CIFAR-10; (b) change of AUROC throughout the training on CIFAR-10; (c) change of AUPR throughout the training on CIFAR-10.



(a) First Layer



(b) 50th Layer



(c) Last Layer (FC)

Figure 11: Histograms of different layers for the original model, pruned model, and UMAP model. The model is DenseNet-101 with a prune rate of 50%. (a) the histogram of the first convolution layer; (b) the histogram of the 50th convolution layer; (c) the histogram of the last (fully connected) layer.