EAT: Towards Long-Tailed Out-of-Distribution Detection

Tong Wei, Bo-Lin Wang, Min-Ling Zhang

School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China
{weit, wangbl, zhangml}@seu.edu.cn

Abstract

Despite recent advancements in out-of-distribution (OOD) detection, most current studies assume a class-balanced in-distribution training dataset, which is rarely the case in real-world scenarios. This paper addresses the challenging task of long-tailed OOD detection, where the in-distribution data follows a long-tailed class distribution. The main difficulty lies in distinguishing OOD data from samples belonging to the tail classes, as the ability of a classifier to detect OOD instances is not strongly correlated with its accuracy on the in-distribution classes. To overcome this issue, we propose two simple ideas: (1) Expanding the in-distribution class space by introducing multiple abstention classes. This approach allows us to build a detector with clear decision boundaries by training on OOD data using *virtual labels*. (2) Augmenting the context-limited tail classes by overlaying images onto the context-rich OOD data. This technique encourages the model to pay more attention to the discriminative features of the tail classes. We provide a clue for separating in-distribution and OOD data by analyzing gradient noise. Through extensive experiments, we demonstrate that our method outperforms the current state-of-the-art on various benchmark datasets. Moreover, our method can be used as an add-on for existing long-tail learning approaches, significantly enhancing their OOD detection performance.

Introduction

Deep neural networks (DNNs) can achieve high performance in various real-world applications by training on large-scale and well-annotated datasets. Most supervised learning literature makes a common assumption that the training and test data have the same distribution. However, DNNs in deployment often encounter data from an unknown distribution, and it has been shown that DNNs tend to produce wrong predictions on anonymous, or out-of-distribution (OOD) test data with high confidence (Hendrycks and Gimpel 2017; Liang, Li, and Srikant 2018; Hein, Andriushchenko, and Bitterwolf 2019), which can result in severe mistakes in practice.

Recently, OOD detection, which aims to reject OOD test data without classifying them as in-distribution labels, has caught great attention. Existing state-of-the-art OOD detectors achieve huge success by maximizing the predictive uncertainty (Hendrycks, Mazeika, and Dietterich 2019; Meinke and Hein 2020), energy function (Liu et al. 2020), and abstention class confidence (Mohseni et al. 2020; Chen et al. 2021) for OOD data. However, these approaches assume that the in-distribution data is class-balanced, which is usually violated in real-world tasks (Van Horn and Perona 2017; Liu et al. 2019; Cui et al. 2019). In this paper, we consider that the in-distribution training data follows a long-tailed class distribution. Under this setup, directly combining existing OOD detectors with long-tailed learning methods still leads to unsatisfactory performance (Wang et al. 2022). So, a natural question is raised:

Is it possible to effectively distinguish OOD data from tail-class samples?

To answer this question, we propose a novel framework, EAT, which is composed of two key ingredients: (1) *dynamic virtual labels*, which expand the classification space with abstention classes for OOD data and are dynamically assigned by the model in the training process. EAT classifies OOD samples into abstention OOD classes rather than imposing uniform predictive probabilities over inlier classes such as in OE (Hendrycks, Mazeika, and Dietterich 2019), Energy (Liu et al. 2020), and PASCL (Wang et al. 2022). This step is critical because inherent similar OOD samples can be pushed closer if they are classified as an identical OOD class, and the decision boundary between inlier data and OOD data will be clearer. (2) *tail class augmentation*, which augments the tail-class images by pasting them onto the context-rich OOD images to force the model to focus on the foreground objects. Precisely, given an original image from the tail class, it is cropped in various sizes and pasted onto images from OOD data. Then, we can create tail-class images with more diverse contexts by changing the background. The generalization for tail classes can be significantly improved.

To further enhance the classification of inlier data, we propose a method that involves fine-tuning the classifiers exclusively using inlier data. This fine-tuning process employs a class-balanced loss function for a few iterations. Additionally, we illustrate that our method can be seamlessly integrated with existing long-tail learning approaches, leading to a significant improvement in their OOD detection performance. This is evident from the results presented in Table 6, where our method acts as a valuable plugin to boost the performance of these approaches. These findings contradict the argument put forth by previous work (Vaze et al. 2022) that a classifier performing well on in-distribution data would automatically excel as an OOD detector.

The key **contributions** of this paper are summarized as follows: (1) We tackle the challenging and under-explored problem of long-tailed OOD detection. This problem poses unique difficulties and requires innovative solutions. (2) We propose a novel approach to train OOD data using virtual labels, presenting an alternative to the outlier exposure method specifically designed for long-tailed data. Furthermore, we provide insights into the impact of virtual labels by examining gradient noise, deepening our understanding of their effectiveness. (3) Through extensive experiments conducted on various datasets, we empirically validate the effectiveness of our proposed method. Our results demonstrate an average boost of 2.0% AUROC and 2.9% inlier classification accuracy compared to the previous state-of-the-art method. (4) Our method serves as a versatile add-on for mainstream long-tailed learning methods, significantly enhancing their performance in detecting OOD samples. Importantly, our findings challenge the notion that a strong inlier classifier necessarily implies good OOD detection performance.

Related Work

OOD detection As a representative approach, Outlier Exposure (OE) proposes maximizing the OOD data's predictive uncertainty as a complementary objective for the in-distribution classification loss. Further, Energy (Liu et al. 2020) improves OE by introducing the energy function as a regularization term and detects OOD samples according to their energy scores. Conversely, SOFL (Mohseni et al. 2020) and ATOM (Chen et al. 2021) attempt to classify OOD samples into abstention classes while in-distribution samples are classified into their true classes. Then, OOD data can be identified according to the model's outputs on abstention classes. Although existing OOD detectors can achieve high performance, they are typically trained on class-balanced in-distribution datasets and cannot be directly applied to long-tailed tasks.

Long-tailed learning Existing approaches to long-tailed learning can be roughly categorized into three types by modifying: (1) the inputs to a model by re-balancing the training data (He and Garcia 2009; Liu et al. 2019; Zhou et al. 2020); (2) the outputs of a model, for example by posthoc adjustment of the classifier (Kang et al. 2020; Menon et al. 2021; Tang, Huang, and Zhang 2020) and (3) the internals of a model by modifying the loss function (Cui et al. 2019; Cao et al. 2019; Jamal et al. 2020; Ren et al. 2020). Recently, (Yang and Xu 2020) and (Wei et al. 2022) propose using OOD data to improve the performance of long-tailed learning. However, it is noted that these approaches are designed to boost the in-distribution classification performance and cannot be directly employed to detect OOD data.

Long-tailed OOD detection Recently, long-tailed OOD detection has received more and more attention, and several approaches have been proposed to tackle this challenging problem. PASCL (Wang et al. 2022) optimizes a contrastive objective between tail class samples and OOD data to push each other away in the latent representation space, which can boost the performance of OOD detection. Further, it minimizes the logit adjustment loss to yield a class-balanced performance of inlier classification. HOD (Roy et al. 2022) studies a long-tail OOD detection problem in medical image analysis, which directly trains a binary classifier to discriminate in-distribution data and OOD data. However, HOD assumes that the OOD data is labeled, while we do not make this assumption and only leverage unlabeled OOD data to aid the detection performance. OLTR (Liu et al. 2019) formally studies the OOD detection task in long-tailed learning. It detects OOD inputs in the latent representation space according to the minimum distance between them and the centroids of in-distribution classes. Although OLTR outperforms several OOD detectors such as MSP (Hendrycks and Gimpel 2017), it is outperformed by the state-of-the-art OOD detection methods, suggesting that there remains room for improvement.

The Proposed Approach

Overview

We follow the popular training objective of existing state-of-the-art OOD detection methods, which train the model using both in-distribution data and unlabeled OOD data. Let $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{out}}$ denote an in-distribution training set and an unlabeled OOD training set, respectively. Note that $\mathcal{D}_{\text{in}}$ follows a long-tailed class distribution in our setup. The training loss function of many existing OOD detection methods (e.g., OE, EnergyOE, ATOM, and PASCL) is defined as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{in}} + \lambda \cdot \mathcal{L}_{\text{out}}, \tag{1}$$

where $\mathcal{L}_{\text{in}}$ is the inlier classification loss, $\mathcal{L}_{\text{out}}$ is the outlier detection loss, and λ is a trade-off hyperparameter. Typically, we choose to optimize the standard cross entropy loss (denoted by ℓ) for the inlier classification task:

$$\begin{aligned}\mathcal{L}_{\text{in}} &= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\text{in}}}[\ell(f(\boldsymbol{x}), y)] \\ &= \log[1 + \sum_{y'\neq y} e^{(f_{y'}(\boldsymbol{x}) - f_y(\boldsymbol{x}))}]\end{aligned} \tag{2}$$

Here, $f_y(\boldsymbol{x})$ represents the predicted logit corresponding to label y. For OOD detection, we propose using k abstention classes. The training outlier data is assigned to abstention classes by generating "virtual" labels by the model, and virtual labels may change through training iterations. With this, the training objective for outliers is defined as:

$$\begin{aligned}\mathcal{L}_{\text{out}} &= \mathbb{E}_{\widetilde{\boldsymbol{x}}\sim\mathcal{D}_{\text{out}}}[\ell(f(\widetilde{\boldsymbol{x}}), \widetilde{y})] \\ &= \log[1 + \sum_{y'\neq\widetilde{y}} e^{(f_{y'}(\widetilde{\boldsymbol{x}}) - f_{\widetilde{y}}(\widetilde{\boldsymbol{x}}))}] \\ &\text{s.t.} \quad \widetilde{y} = \arg\max_{c\in[C+1, C+k]} f_c(\widetilde{\boldsymbol{x}})\end{aligned} \tag{3}$$

where $\widetilde{y}$ is the virtual label of outlier sample $\widetilde{\boldsymbol{x}}$. Note that our treatment for training outlier data differs from existing methods, including OE, Energy, and PASCL. They attempt to maximize the predictive uncertainties of outliers. We demonstrate that our approach achieves significantly better results in the experiments by introducing multiple abstention classes. The proposed approach is detailed below.

OOD Samples with Dynamic Virtual Labels

The approach of using abstention OOD classes is motivated by recent works (Abdelzad et al. 2019; Chen et al. 2021; Vernekar et al. 2019) which propose to add a single abstention class for all outlier data. Although this is shown to be effective compared to the outlier exposure method (Hendrycks, Mazeika, and Dietterich 2019), fitting a heterogeneous outlier set to a single class is challenging and problematic. One natural mitigation strategy here is to assign multiple abstention classes as possible outputs, which essentially turns the C-class classification into a $(C+k)$-class classification problem. Here, we denote C as the number of inlier classes and k as the number of abstention classes added for outliers. Taking CIFAR100-LT as an example, if we use an additional $k = 30$ classes for fitting outliers, the number of neurons in the final fully-connected layer will be 130.

Ultimately, we want our model to classify unseen outliers in the test set into those k abstention classes. This can be achieved by encouraging the model to learn a structured decision boundary for the inliers *vs.* outliers. However, the ground-truth labels for training outlier data are not accessible. Thus, we propose generating virtual labels for the outliers so that the model learns to distinguish them from inliers. Towards this end, we take the predictions of the immediate model as the virtual labels at each training iteration, also known as self-labelling. The model is trained to predict virtual labels by minimizing the cross-entropy loss at the next iteration. The generation of virtual labels coincides with the self-training process, which is a popular framework in semi-supervised learning. At test time, the sum of probabilities for the k abstention classes indicating the OOD score is used. This is because the abstention classes are meaningless and virtual labels do not correspond to their ground-truth labels.

Mathmatical Interpretation Exploring the reason behind OOD samples yielding higher scores than in-distribution samples is an intriguing endeavor. One way to comprehend the impact of virtual labels is through the lens of noise in loss gradients (Wei et al. 2021). We define the trainable parameter of model f as $\boldsymbol{\theta} \in \mathbb{R}^p$. By calculating the gradient of the loss function with respect to $\boldsymbol{\theta}$ and updating the parameter accordingly, we gain insight into this phenomenon. Specifically, we represent the output probabilities for an in-distribution sample $\boldsymbol{x}$ and an OOD sample $\widetilde{\boldsymbol{x}}$ as $\boldsymbol{z} = \mathrm{Softmax}(f(\boldsymbol{x}))$ and $\widetilde{\boldsymbol{z}} = \mathrm{Softmax}(f(\widetilde{\boldsymbol{x}}))$ respectively.

Proposition 1. *For the cross-entropy loss, Eq.* (3) *induces gradient noise* $\boldsymbol{g} = -\frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}$ *on* $\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{z}, y)$*, s.t.,* $\boldsymbol{g} \in \mathbb{R}^p, j = \arg\max_{j \in [C+1, C+k]} \widetilde{\boldsymbol{z}}$*. While each OOD sample in OE (Hendrycks, Mazeika, and Dietterich 2019) induces gradient noise* $\boldsymbol{g}' = -\frac{1}{C} \sum_{j=1}^{C} \frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}$ *on* $\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{z}, y)$*, where* $\frac{\cdot}{\widetilde{\boldsymbol{z}}_j}$ *denotes the element-wise division.*

Remark. The detailed proof for the following proposition can be found in the supplementary material. We first draw the conclusion that our proposed virtual labeling induces gradient noise of $\boldsymbol{g} = -\frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}} j}$ where $j \in [C+1, C+k]$ is the virtual label for an OOD sample. On the contrary, previous method OE induces gradient noise of $\boldsymbol{g}' = -\frac{1}{C} \sum_{i=1}^{C} \frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_i}{\widetilde{\boldsymbol{z}} i}$. Therefore, the main advantage of our approach yields gradient noise with dynamic direction depending on the virtual label of each OOD sample, which helps escape local minima during optimization. However, OE induces constant gradient noise so that the optimization of the model always follows the direction of gradient descent. Furthermore, the gradient noise induced by our approach helps the model to produce more conservative in-distribution class (i.e., $[1, C]$) predictions on OOD samples than OE. This is because of the nature of virtual labels which encourages the model to produce confident predictions on virtual classes, i.e., $[C+1, C+k]$.

Context-rich Tail Class Augmentation In our pursuit to enhance generalization, we go beyond the utilization of virtual labels to amplify the distinction between in-distribution and OOD samples. We additionally harness OOD samples to augment the tail classes, leading to improved performance. Our approach involves the implementation of an image-mixing data augmentation technique called CutMix (Yun et al. 2019), which enables us to generate training samples specifically tailored for the tail class. The core concept revolves around leveraging the context-rich nature of the head class and outlier images as backgrounds to create diverse and enriched tail samples. Given a tail-class image $\boldsymbol{x}_f$, we combine it with a randomly selected head-class or OOD image represented as $\boldsymbol{x}_b$. This merging operation is referred to as the CutMix operator and is defined as follows:

$$\boldsymbol{x}^{\mathrm{b} \odot \mathrm{f}} = \mathbf{M} \odot \boldsymbol{x}^{\mathrm{b}} + (\mathbf{1} - \mathbf{M}) \odot \boldsymbol{x}^{\mathrm{f}} \tag{4}$$

In this context, we designate $\boldsymbol{x}^{\mathrm{b}}$ as the background image and $\boldsymbol{x}^{\mathrm{f}}$ as the foreground image. A binary mask $\mathbf{M} \in 0, 1^{W \times H}$ is employed to indicate the areas to preserve as background. Correspondingly, $(\mathbf{1} - \mathbf{M})$ selects the patch from the foreground image to be pasted onto the background image. Here, $\mathbf{1}$ represents a matrix filled with ones, and $\odot$ denotes element-wise multiplication. In order to address the limited availability of data for tail classes, we assume that the composite image $\boldsymbol{x}^{\mathrm{b}} \odot \boldsymbol{x}^{\mathrm{f}}$ carries the same label as the foreground image $\boldsymbol{x}^{\mathrm{f}}$, i.e., $y^{\mathrm{b} \odot \mathrm{f}} = y^{\mathrm{f}}$. However, it is important to note that this approach can introduce label noise during training. Therefore, we assign lower sample weights to the generated tail-class images to mitigate the adverse impact.

Tailored CutMix offers two notable advantages for both outlier detection and inlier classification. Firstly, by using diverse OOD images as backgrounds, the model is encouraged to differentiate between tail-class images and OOD data based on foreground objects rather than image backgrounds. This aids in enhancing the model's ability to identify and distinguish outliers effectively. Secondly, the inclusion of head-class and OOD data through mixing increases the frequency of tail classes, leading to a more balanced training set. This improved class balance contributes to enhanced generalization capabilities.

It is worth noting that the study conducted by (Park et al. 2022) also incorporates CutMix to generate tail-class samples. However, their approach differs in that they sample image pairs from the original long-tailed data distribution and a tail-class-weighted distribution. Furthermore, their study focuses on improving inlier prediction accuracy rather than OOD

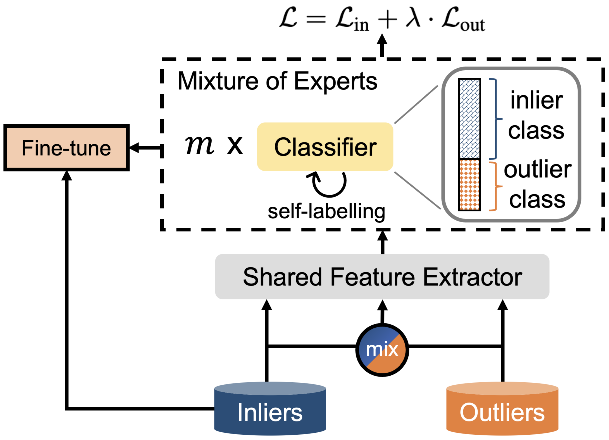

Figure 1: Overview of EAT framework.

detection. As far as our knowledge extends, we are the first to adapt CutMix specifically for long-tailed OOD detection, distinguishing our work in this area.

Improving OOD Separation

To amplify both outlier detection and inlier classification performance, we employ a mixture of experts by integrating multiple classifiers that share a common feature extractor. By training an ensemble of m members with random initializations, we optimize the sum of loss functions for these classifiers, aiming to achieve superior results.

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{m} (\mathcal{L}_{\text{in}}^{(i)} + \lambda \cdot \mathcal{L}_{\text{out}}^{(i)}) \tag{5}$$

Given an input $\boldsymbol{x}$ at test time, we use the average predictions of ensemble members as the OOD score:

$$G(\boldsymbol{x}) = \frac{1}{m} \sum_{i=1}^{m} \sum_{j=C+1}^{C+k} z_j^{(i)}, \tag{6}$$

where $\boldsymbol{z}^{(i)} = \text{Softmax}(f^{(i)}(\boldsymbol{x}))$. We choose $m = 3$ in our experiments. If $\boldsymbol{x}$ is not deemed as an OOD input, the prediction will be $\arg\max_{1 \leq c \leq C} \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{z}^{(i)}$.

It is important to note that the performance improvement achieved by deep ensembles relies on the diversity introduced through random initialization of network parameters. In our specific setup, since we employ a shared feature extractor, random initialization is applied solely to the parameters of the last layer. To enhance diversity further, we train ensemble models with virtual labels generated by each classifier. This means that the virtual label for a given sample $\boldsymbol{x}$ is obtained by selecting $\tilde{y} = \arg\max_{c \in [C+1, C+k]} f_c^{(i)}(\boldsymbol{x})$ for the i-th classifier. The overall framework of our approach is depicted in Figure 1.

Model Fine-tuning

After training a multi-branch model, we proceed to fine-tune the classifiers using training inlier data, aiming to enhance the classification performance. It is worth mentioning that the two-stage approach, involving representation learning followed by classifier learning, is commonly employed in the field of long-tailed learning. Prominent examples include decoupling (Kang et al. 2020), BBN (Zhou et al. 2020), and MisLAS (Zhong et al. 2021). In this article, we implicitly explore the advantages of utilizing OOD data for representation learning by optimizing supervised objectives during model training. Furthermore, we keep the feature extractor fixed and refine the classifiers for a few iterations to improve inlier classification performance. During the fine-tuning process, we employ the logits adjustment (LA) loss (Menon et al. 2021) to guide the training.

$$\ell_{\text{LA}}(y, f(\boldsymbol{x})) = \log[1 + \textstyle\sum_{y' \neq y} e^{\triangle_{yy'}} \cdot e^{(f_{y'}(\boldsymbol{x}) - f_y(\boldsymbol{x}))}] \tag{7}$$

Here, the pairwise label margins $\triangle_{yy'} = \log \frac{\pi_y}{\pi_{y'}}$ represents the desired gap between predictive confidence for y and y' depending on the number of each class. π_y denotes the class prior of class y in the training inlier data. We will empirically show that fine-tuning can introduce not only large improvements for the inlier classification task but also boost the performance of OOD detection.

Experiments

Experiment Settings

We verify our approach on commonly used datasets in comparison with the existing state-of-the-art. CIFAR10-LT, CIFAR100-LT (Cao et al. 2019), and ImageNet-LT (Liu et al. 2019) are used as in-distribution training sets ($\mathcal{D}_{\text{in}}$). The standard CIFAR10, CIFAR100, and ImageNet test sets are used as in-distribution test sets ($\mathcal{D}_{\text{in}}^{\text{test}}$). Following (Wang et al. 2022), we set the default imbalance ratio to 100 for CIFAR10-LT and CIFAR100-LT during training.

OOD datasets for CIFAR-LT We employ 300 thousand samples from TinyImages80M (Torralba, Fergus, and Freeman 2008) as the OOD training images for CIFAT10-LT and CIFAR100-LT following (Hendrycks, Mazeika, and Dietterich 2019; Wang et al. 2022). Of those, 80 Million Tiny Images is a large-scale, diverse dataset of 32 × 32 natural images. The 300 thousand samples are selected from the 80 Million Tiny Images by (Hendrycks, Mazeika, and Dietterich 2019), not intersected with the CIFAR datasets. For OOD test data, we use Textures (Cimpoi et al. 2014), SVHN (Netzer et al. 2011), Tiny ImageNet (Le and Yang 2015), LSUN (Yu et al. 2015), and Places365 (Zhou et al. 2017) as $\mathcal{D}_{\text{out}}^{\text{test}}$. We use CIFAR-100 as a $\mathcal{D}_{\text{out}}^{\text{test}}$ for CIFAR10-LT and vice-versa.

OOD datasets for ImageNet-LT We use a specifically designed $\mathcal{D}_{\text{out}}$ called ImageNet-Extra following (Wang et al. 2022). ImageNet-Extra has $517,711$ images belonging to 500 classes randomly sampled from ImageNet-22k (Deng et al. 2009), but not overlapping with the $1,000$ in-distribution classes in ImageNet-LT. For $\mathcal{D}_{\text{out}}^{\text{test}}$, we use ImageNet-1k-OOD constructed by (Wang et al. 2022), which has $50,000$ OOD test images from $1,000$ classes randomly selected from ImageNet-22k (with 50 images in each class). Considering the fairness of OOD detection, it has the same size as the in-distribution test set. The $1,000$ classes in ImageNet-1k-OOD

$\mathcal{D}_{\text{out}}^{\text{test}}$	**Method**	**AUROC**	**AUPR**	**FPR95**	**ACC95**
Texture	OE	92.59$_{\pm0.42}$	83.32$_{\pm1.67}$	25.10$_{\pm1.08}$	84.52$_{\pm0.76}$
	PASCL	93.16$_{\pm0.37}$	84.80$_{\pm1.50}$	23.26$_{\pm0.91}$	85.86$_{\pm0.72}$
	Ours	**95.44**$_{\pm0.46}$	**92.28**$_{\pm0.95}$	**21.50**$_{\pm1.50}$	**87.00**$_{\pm0.66}$
SVHN	OE	95.10$_{\pm1.01}$	97.14$_{\pm0.81}$	16.15$_{\pm1.52}$	81.33$_{\pm0.81}$
	PASCL	96.63$_{\pm0.90}$	98.06$_{\pm0.56}$	12.18$_{\pm3.33}$	82.72$_{\pm1.51}$
	Ours	**97.92**$_{\pm0.36}$	**99.06**$_{\pm0.20}$	**9.87**$_{\pm2.06}$	**84.39**$_{\pm0.51}$
CIFAR100	OE	83.40$_{\pm0.30}$	80.93$_{\pm0.57}$	56.96$_{\pm0.91}$	94.56$_{\pm0.57}$
	PASCL	84.43$_{\pm0.23}$	82.99$_{\pm0.48}$	57.27$_{\pm0.88}$	94.48$_{\pm0.31}$
	Ours	**85.93**$_{\pm0.15}$	**86.10**$_{\pm0.35}$	**54.13**$_{\pm0.63}$	**95.81**$_{\pm0.41}$
Tiny ImageNet	OE	86.14$_{\pm0.29}$	79.33$_{\pm0.65}$	47.78$_{\pm0.72}$	91.19$_{\pm0.33}$
	PASCL	87.14$_{\pm0.18}$	81.54$_{\pm0.38}$	47.69$_{\pm0.59}$	91.20$_{\pm0.35}$
	Ours	**89.11**$_{\pm0.34}$	**85.43**$_{\pm0.58}$	**41.75**$_{\pm0.68}$	**91.67**$_{\pm0.65}$
LSUN	OE	91.35$_{\pm0.23}$	87.62$_{\pm0.82}$	27.86$_{\pm0.68}$	85.49$_{\pm0.69}$
	PASCL	93.17$_{\pm0.15}$	91.76$_{\pm0.53}$	26.40$_{\pm1.00}$	86.67$_{\pm0.90}$
	Ours	**95.13**$_{\pm0.43}$	**94.12**$_{\pm0.61}$	**19.72**$_{\pm1.61}$	**86.68**$_{\pm0.64}$
Places365	OE	90.07$_{\pm0.26}$	95.15$_{\pm0.24}$	34.04$_{\pm0.91}$	87.07$_{\pm0.53}$
	PASCL	91.43$_{\pm0.17}$	96.28$_{\pm0.14}$	33.40$_{\pm0.88}$	**87.87**$_{\pm0.71}$
	Ours	**93.68**$_{\pm0.27}$	**97.42**$_{\pm0.14}$	**26.03**$_{\pm0.92}$	87.64$_{\pm0.68}$
Average	OE	89.77$_{\pm0.27}$	87.25$_{\pm0.61}$	34.65$_{\pm0.46}$	87.36$_{\pm0.51}$
	PASCL	90.99$_{\pm0.19}$	89.24$_{\pm0.34}$	33.36$_{\pm0.79}$	88.13$_{\pm0.56}$
	Ours	**92.87**$_{\pm0.33}$	**92.40**$_{\pm0.47}$	**28.83**$_{\pm1.23}$	**88.86**$_{\pm0.59}$

(a) OOD detection results and in-distribution classification results in terms of ACC95.

Method	**ACC@FPRn (↑)**			
	0	0.001	0.01	0.1
OE	73.54$_{\pm0.77}$	73.90$_{\pm0.77}$	74.46$_{\pm0.81}$	78.88$_{\pm0.66}$
PASCL	77.08$_{\pm1.01}$	77.13$_{\pm1.02}$	77.64$_{\pm0.99}$	81.96$_{\pm0.85}$
Ours	**81.31**$_{\pm0.26}$	**81.36**$_{\pm0.25}$	**81.81**$_{\pm0.26}$	**84.40**$_{\pm0.28}$

(b) In-distribution classification results in terms of ACC@FPRn.

Method	**AUROC (↑)**	**AUPR (↑)**	**FPR95 (↓)**	**ACC (↑)**
ST (MSP)	72.28	70.27	66.07	72.34
OECC	87.28	86.29	45.24	60.16
EnergyOE	89.31	88.92	40.88	74.68
OE	89.77$_{\pm0.27}$	87.25$_{\pm0.61}$	34.65$_{\pm0.46}$	73.84$_{\pm0.77}$
PASCL	$\underline{90.99}_{\pm0.19}$	$\underline{89.24}_{\pm0.34}$	$\underline{33.36}_{\pm0.79}$	$\underline{77.08}_{\pm1.01}$
Ours	**92.87**$_{\pm0.33}$	**92.40**$_{\pm0.47}$	**28.83**$_{\pm1.23}$	**81.31**$_{\pm0.26}$

(c) Comparison with other methods.

Table 1: Results on CIFAR10-LT using ResNet18. The best results are shown in bold. Mean and standard deviation over six random runs are reported. "Average" means the results averaged across six different $\mathcal{D}_{\text{out}}^{\text{test}}$ sets.

are not intersecting either the $1,000$ in-distribution classes in ImageNet-LT or the 500 OOD training classes in ImageNet-Extra. To ensure the rigor of the experiment, ImageNet-LT $\mathcal{D}_{\text{train}}^{\text{in}}$, ImageNet-Extra $\mathcal{D}_{\text{train}}^{\text{out}}$, ImageNet-1k-OOD $\mathcal{D}_{\text{test}}^{\text{out}}$, and ImageNet $\mathcal{D}_{\text{test}}^{\text{in}}$ are orthogonal.

Evaluation measures Following (Hendrycks, Mazeika, and Dietterich 2019; Mohseni et al. 2020; Yang et al. 2021; Wang et al. 2022), we use the below evaluation measures:

- **AUROC** (↑): The area under the receiver operating characteristic curve. AUROC means whether the OOD detector can appropriately rank OOD samples higher in-distribution samples.
- **AUPR** (↑): The area under the precision-recall curve, which is also known as the average precision overall recall value.
- **FPR@TPR**n (↓): The false positive rate (FPR) when n (in percentage) OOD samples have been successfully detected (i.e., when the true positive rate (TPR) is n). (Hendrycks and Gimpel 2017) have primarily used the measure FPR95, also known as FPR@TPR95%.
- **ACC@TPR**n (↑): The classification accuracy on the remaining in-distribution data when n (in percentage) OOD samples have been successfully detected. The term ACC@TPR95% is shortened to ACC95.
- **ACC@FPR**n (↑): The classification accuracy on the remaining in-distribution data when n (in percentage) in-distribution samples are mistakenly detected as OOD. The accuracy on the overall in-distribution test set is known as ACC@FPR0, or simply ACC.

Model Configuration The current best long-tail OOD detection method is PASCL, and before that it is OE, so we mainly compare the experimental results with these two baseline methods. For experiments on CIFAR10 and CIFAR100, we use the ResNet18 (He et al. 2016) following (Yang et al. 2021). For experiments on CIFAR10-LT and CIFAR100-LT, we train the model for 180 epochs using Adam (Kingma and Ba 2014) optimizer with initial learning rate 1×10^{-3} and batch size 128. We decay the learning rate to 0 using a cosine annealing learning rate scheduler (Loshchilov and Hutter 2016). For fine-tuning, we fine-tune the classifier for 10 epochs using Adam optimizer with an initial learning rate 5×10^{-4}. Other hyper-parameters are the same as in classifier layer fine-tuning. For experiments on ImageNet-LT, we follow the settings in (Wang et al. 2021) and use ResNet50 (He et al. 2016). We train the main branch for 60 epochs using SGD optimizer with an initial learning rate of 0.1 and batch size of 64. We fine-tune the classifier for the 1 epoch using SGD optimizer with an initial learning rate of 0.01. In all experiments, we set $\lambda = 0.05$, and the weights for generated tail class samples are set to 0.05 for EAT. For the number of abstention classes, we set $k = 3$ on CIFAR10-LT, $k = 30$ on CIFAR100-LT and ImageNet-LT. For other hyper-parameters in the baseline methods, we use the suggested values in their original papers.

Main Results

Table 1, Table 2, and Table 4 report the results for CIFAR10-LT, CIFAR100-LT, and ImageNet-LT datasets, respectively. For fair comparison, results of existing methods are directly borrowed from (Wang et al. 2022). There are three sub-tables in Table 1 and Table 2: since performance measures may differ across different $\mathcal{D}_{\text{out}}^{\text{test}}$ datasets, we report AUROC, AUPR, FPR95, and ACC95 on each $\mathcal{D}_{\text{out}}^{\text{test}}$ as well as the average values across six $\mathcal{D}_{\text{out}}^{\text{test}}$ datasets in sub-table (a). In sub-table (b), we compare ACC@FPRn with various n values that are independent of $\mathcal{D}_{\text{out}}^{\text{test}}$. Finally, we put together four main performance measures in terms of both outlier detection and inlier classification in sub-table (c).

$\mathcal{D}_{\text{out}}^{\text{test}}$	**Method**	**AUROC**	**AUPR**	**FPR95**	**ACC95**
Texture	OE	76.71$_{\pm 1.20}$	58.79$_{\pm 1.39}$	68.28$_{\pm 1.53}$	71.43$_{\pm 1.58}$
	PASCL	76.01$_{\pm 0.66}$	58.12$_{\pm 1.06}$	**67.43**$_{\pm 1.93}$	73.11$_{\pm 1.55}$
	Ours	**80.27**$_{\pm 0.76}$	**71.76**$_{\pm 1.56}$	67.53$_{\pm 0.64}$	**73.76**$_{\pm 0.75}$
SVHN	OE	77.61$_{\pm 3.26}$	86.82$_{\pm 2.50}$	58.04$_{\pm 4.82}$	64.27$_{\pm 3.26}$
	PASCL	80.19$_{\pm 2.19}$	88.49$_{\pm 1.59}$	53.45$_{\pm 3.60}$	**64.50**$_{\pm 1.87}$
	Ours	**83.11**$_{\pm 2.83}$	**89.71**$_{\pm 2.08}$	**47.78**$_{\pm 4.87}$	61.67$_{\pm 2.65}$
CIFAR10	OE	62.23$_{\pm 0.30}$	**57.57**$_{\pm 0.34}$	80.64$_{\pm 0.98}$	**82.67**$_{\pm 0.99}$
	PASCL	**62.33**$_{\pm 0.38}$	57.14$_{\pm 0.20}$	79.55$_{\pm 0.84}$	82.30$_{\pm 1.07}$
	Ours	61.62$_{\pm 0.47}$	55.30$_{\pm 0.54}$	**77.97**$_{\pm 0.77}$	82.61$_{\pm 0.61}$
Tiny ImageNet	OE	68.04$_{\pm 0.37}$	51.66$_{\pm 0.51}$	76.66$_{\pm 0.47}$	76.22$_{\pm 0.61}$
	PASCL	68.20$_{\pm 0.37}$	51.53$_{\pm 0.42}$	76.11$_{\pm 0.80}$	**77.56**$_{\pm 1.15}$
	Ours	**68.34**$_{\pm 0.28}$	**52.79**$_{\pm 0.25}$	**74.89**$_{\pm 0.49}$	77.07$_{\pm 0.39}$
LSUN	OE	77.10$_{\pm 0.64}$	61.42$_{\pm 0.99}$	63.98$_{\pm 1.38}$	65.64$_{\pm 1.03}$
	PASCL	77.19$_{\pm 0.44}$	61.27$_{\pm 0.72}$	63.31$_{\pm 0.87}$	**68.05**$_{\pm 1.24}$
	Ours	**81.09**$_{\pm 0.32}$	**67.46**$_{\pm 0.64}$	**55.02**$_{\pm 1.20}$	62.07$_{\pm 0.78}$
Places365	OE	75.80$_{\pm 0.45}$	86.68$_{\pm 0.38}$	65.72$_{\pm 0.92}$	67.04$_{\pm 0.49}$
	PASCL	76.02$_{\pm 0.21}$	86.52$_{\pm 0.29}$	64.81$_{\pm 0.27}$	**69.04**$_{\pm 0.90}$
	Ours	**78.28**$_{\pm 0.31}$	**88.20**$_{\pm 0.20}$	**60.85**$_{\pm 0.69}$	66.15$_{\pm 0.68}$
Average	OE	72.91$_{\pm 0.68}$	67.16$_{\pm 0.57}$	68.89$_{\pm 1.07}$	71.21$_{\pm 0.84}$
	PASCL	73.32$_{\pm 0.32}$	67.18$_{\pm 0.10}$	67.44$_{\pm 0.58}$	**72.43**$_{\pm 0.66}$
	Ours	**75.45**$_{\pm 0.83}$	**70.87**$_{\pm 0.88}$	**64.01**$_{\pm 1.44}$	70.55$_{\pm 0.98}$

(a) In-distribution classification results in terms of ACC@FPRn.

Method	**ACC@FPRn (↑)**			
	0	0.001	0.01	0.1
OE	39.04$_{\pm 0.37}$	39.07$_{\pm 0.38}$	39.38$_{\pm 0.38}$	42.40$_{\pm 0.44}$
PASCL	43.10$_{\pm 0.47}$	43.12$_{\pm 0.47}$	43.39$_{\pm 0.48}$	46.14$_{\pm 0.38}$
Ours	**46.23**$_{\pm 0.25}$	**46.24**$_{\pm 0.25}$	**46.38**$_{\pm 0.23}$	**48.39**$_{\pm 0.32}$

(b) OOD detection results and in-distribution classification results in terms of ACC95.

Method	**AUROC** (↑)	**AUPR** (↑)	**FPR95** (↓)	**ACC** (↑)
ST (MSP)	61.00	57.54	82.01	40.97
OECC	70.38	66.87	73.15	32.93
EnergyOE	71.10	67.23	71.78	39.05
OE	72.91$_{\pm 0.68}$	67.16$_{\pm 0.57}$	68.89$_{\pm 1.07}$	39.04$_{\pm 0.37}$
PASCL	$\underline{73.32}_{\pm 0.32}$	$\underline{67.18}_{\pm 0.10}$	$\underline{67.44}_{\pm 0.58}$	$\underline{43.10}_{\pm 0.47}$
Ours	**75.45**$_{\pm 0.83}$	**70.87**$_{\pm 0.88}$	**64.01**$_{\pm 1.44}$	**46.23**$_{\pm 0.25}$

(c) Comparison with other methods.

Table 2: Results on CIFAR100-LT using ResNet18. The best results are shown in bold. Mean and standard deviation over six random runs are reported. "Average" means the results averaged across six different $\mathcal{D}_{\text{out}}^{\text{test}}$ sets.

From the results, we can see that our approach significantly outperforms OE, PASCL, and other baselines. For instance, on CIFAR10-LT, our approach achieves 1.88% AUROC, 3.16% AUPR, 4.53% FPR95, 0.73% ACC95, and 4.23% in-distribution accuracy improvement than PASCL on average. Likewise, on CIFAR100-LT, our approach achieves 2.13% higher AUROC, 3.69% higher AUPR, 3.43% lower FPR95, and 3.13% higher classification accuracy than PASCL on average.

On ImageNet-LT, our approach achieves the best results in seven cases, while the previous state-of-the-art method PASCL performs the best in only one case. Compared with OE, our approach achieves 3.51% higher AUROC, 0.96% higher AUPR, and 9.19% higher in-distribution accuracy.

Method	**ACC (↑)**	
	Head classes	Tail classes
OE	54.29	20.90
PASCL	54.73 (+0.44)	36.26 (+15.36)
Ours	59.46 (+5.17)	34.12 (+13.22)

Table 3: Results on ImageNet-LT.

Improvements on head and tail classes In Table 3, we show the improvements of our method over OE on head and tail in-distribution classes. As we can see, our approach can substantially benefit both the head and tail classes. Compared with PASCL, it is highly biased towards the tail class and the improvement on head classes is marginal, our method achieves a good balance.

Why our method achieves low ACC@TPRn? We show the failure cases. Table 2 and Table 4 show several cases where the baseline is better than our approach concerning ACC@TPRn. We empirically find this is due to more in-distribution samples preserved by our approach than the baselines when a percentage of OOD samples have been successfully detected. As a result, the classification accuracy of the remaining samples may be lower even though our approach correctly classifies more in-distribution samples than baselines. We provide more statistics in the supplementary.

Ablation Study

How do the key components of EAT affect the performance? In Table 5, we study the effects of the four critical components in our EAT approach: (1) virtual labels, (2) classifier fine-tuning, (3) CutMix, and (4) the mixture of experts (MoE), on CIFAR10-LT and CIFAR100-LT datasets. Since the performance of most approaches fluctuates on SVHN, we choose SVHN as $\mathcal{D}_{\text{out}}^{\text{test}}$.

First, on both CIFAR10-LT and CIFAR100-LT datasets, employing the virtual label strategy for OOD data significantly improves OOD detection performance. It improves the AUROC, AUPR, and FPR95 by an average margin of 10%. Note that, although without using virtual labels achieves higher results of ACC95, it is because more in-distribution samples are incorrectly deemed as OOD by the model.

Second, fine-tuning improves the inlier classification while maintaining a competitive OOD detection performance. For instance, the ACC@FPR increases by about 3% on average. Moreover, since we fine-tune the classifiers for only one iteration, it does not introduce much extra computational cost.

Third, we study the role of CutMix. We find it beneficial to use context-rich images as backgrounds to improve the performance of OOD detection and inlier classification. For OOD detection, it improves AUROC and FPR95 by about 4% and 8% on CIFAR100-LT, respectively. For inlier classification, the ACC@FPR is improved by an average of 2% on CIFAR100-LT. Notably, we observe different results on CIFAR100-LT and CIFAR10-LT concerning ACC@FPRn, which is related to the in-distribution accuracy. This may be

Method	AUROC (↑)	AUPR (↑)	FPR@TPRn (↓)				ACC@TPRn (↑)				ACC@FPRn (↑)			
			0.98	0.95	0.90	0.80	0.98	0.95	0.90	0.80	0	0.001	0.01	0.1
ST (MSP)	53.81	51.63	95.38	90.15	83.52	72.97	**96.67**	**92.61**	**87.43**	**77.52**	39.65	39.68	40.00	43.18
OECC	63.07	63.05	**93.15**	**86.90**	78.79	65.23	94.25	88.23	80.12	68.36	38.25	38.28	38.56	41.47
EnergyOE	64.76	64.77	94.15	87.72	78.36	63.71	80.18	74.38	67.65	59.68	38.50	38.52	38.72	40.99
OE	66.33	68.29	95.11	88.22	78.68	65.28	95.46	88.22	78.68	65.28	37.60	37.62	37.79	40.00
PASCL	68.00	**70.15**	94.38	87.53	78.12	62.48	95.69	89.55	80.88	69.60	45.49	45.51	45.62	47.49
Ours	**69.84**	69.25	94.34	87.63	**77.30**	**57.81**	83.22	77.80	70.84	61.49	**46.79**	**46.79**	**46.83**	**48.30**

Table 4: Results on ImageNet-LT. The best and second-best are bolded and underlined, respectively.

$\mathcal{D}_{\text{in}}$	Virtual label	Fine-tuning	CutMix	MoE	AUROC (↑)	AUPR (↑)	FPR95 (↓)	ACC95 (↑)	ACC@FPRn (↑)			
									0	0.001	0.01	0.1
CIFAR10-LT	✗			$***$	84.91	91.75	47.46	**92.91**	77.48	77.52	77.92	81.65
		✗		$***$	96.10	98.02	16.87	83.53	78.49	78.54	79.00	81.85
			✗	$***$	96.62	98.22	14.92	84.79	79.63	79.69	80.19	83.47
				$*$	97.08	98.70	13.86	84.09	80.38	80.41	80.72	83.10
				$**$	97.62	98.93	11.43	83.87	80.49	80.53	80.92	83.49
	EAT				**97.92**	**99.06**	**9.87**	84.39	**81.31**	**81.36**	**81.81**	**84.40**
CIFAR100-LT	✗			$***$	74.04	84.99	63.58	**73.98**	**46.68**	**46.70**	**46.91**	**49.50**
		✗		$***$	81.77	89.67	54.53	64.47	43.34	43.36	43.59	46.00
			✗	$***$	79.52	88.11	55.89	64.30	43.93	43.94	44.10	46.24
				$*$	80.70	88.27	52.86	64.30	45.82	45.84	45.97	47.94
				$**$	82.13	89.01	50.51	63.18	46.32	46.32	46.48	48.57
	EAT				**83.11**	**89.71**	**47.78**	61.67	46.23	46.24	46.38	48.39

Table 5: The impact of key ingredients for EAT. Experiments are conducted on CIFAR10-LT and CIFAR100-LT ($\rho = 100$). SVHN is used as $\mathcal{D}_{\text{out}}^{\text{test}}$. The number of $*$ denotes the ensemble size.

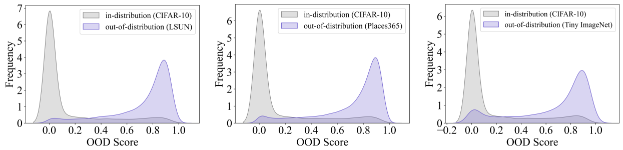

Figure 2: Distribution of OOD scores from our model. The CIFAR10 is used as the in-distribution dataset, and the other six are OOD datasets. It shows that both in-distribution data and OOD data naturally form smooth distributions.

related to the value of k because a larger k means that more abstention class heads need to be learned, resulting in a little effect on ID accuracy. Furthermore, Figure 2 illustrates the distribution of OOD scores on the other three OOD datasets, and our model distinguishes OOD data from in-distribution data with a clear decision boundary.

Method	ACC (↑)	AUROC (↑)	AUPR (↑)	FPR95 (↓)
RIDE (Wang et al. 2021)	**48.49**	66.18	62.13	77.73
RIDE+Ours	47.94	**72.41**	**69.37**	**71.99**
GLMC (Du et al. 2023)	**54.51**	65.01	62.01	79.65
GLMC+Ours	52.30	**73.07**	**67.14**	**69.31**

Table 6: Combining with other methods. The experiment is conducted on CIFAR-100 (in-distribution) dataset and six OOD datasets.

A Good Closed-Set Classifier is All You Need? An interesting finding from previous work (Vaze et al. 2022) suggests that utilizing the maximum logit score rule with a highly accurate in-distribution classifier can outperform many well-designed OOD detectors. However, we sought to investigate whether this holds true in the context of long-tailed tasks. To validate this, we conducted experiments involving two sophisticated long-tail learning methods, namely RIDE (Wang et al. 2021) and GLMC (Du et al. 2023). We find that their OOD detection performance lagged significantly behind that of our proposed method, providing evidence that a strong classifier alone is insufficient for effective long-tailed OOD detection. Furthermore, when we combined our method with RIDE and GLMC, as shown in Table 6, we observe a substantial improvement in OOD detection performance with minimal sacrifice in in-distribution classification accuracy, underscoring the versatility of our approach.

Conclusion

In the real-world deployment of machine learning models, test inputs with previously unseen classes are often encountered. For safety, it may be essential to identify such inputs. Moreover, the class-balanced training in-distribution data assumption rarely holds in the wild. This paper proposes a novel framework (EAT) to tackle the long-tailed OOD detection problem. Towards this end, EAT presents several general techniques that can easily be applied to mainstream OOD detectors and long-tail learning methods. First, the abstention OOD classes can be used as an alternative to the outlier exposure method. Second, tail-class augmentation can be employed as a universal add-on for existing methods. Third, the classifier ensembling technique can further boost the performance without introducing much additional computational cost. Finally, we evaluate the proposed method on many commonly used datasets, showing that it consistently outperforms the existing state-of-the-art.

References

Abdelzad, V.; Czarnecki, K.; Salay, R.; Denounden, T.; Vernekar, S.; and Phan, B. 2019. Detecting out-of-distribution inputs in deep neural networks using an early-layer output. *arXiv preprint arXiv:1910.10307*.

Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; and Ma, T. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 1567–1578.

Chen, J.; Li, Y.; Wu, X.; Liang, Y.; and Jha, S. 2021. ATOM: Robustifying out-of-distribution detection using outlier mining. In *ECML*, 430–445.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *CVPR*, 3606–3613.

Cui, Y.; Jia, M.; Lin, T.-Y.; Song, Y.; and Belongie, S. 2019. Class-balanced loss based on effective number of samples. In *CVPR*, 9268–9277.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*, 248–255.

Du, F.; Yang, P.; Jia, Q.; Nan, F.; Chen, X.; and Yang, Y. 2023. Global and Local Mixture Consistency Cumulative Learning for Long-tailed Visual Recognitions. In *CVPR*.

He, H.; and Garcia, E. A. 2009. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9): 1263–1284.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hein, M.; Andriushchenko, M.; and Bitterwolf, J. 2019. Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *CVPR*, 41–50.

Hendrycks, D.; and Gimpel, K. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*.

Hendrycks, D.; Mazeika, M.; and Dietterich, T. 2019. Deep Anomaly Detection with Outlier Exposure. In *ICLR*.

Jamal, M. A.; Brown, M.; Yang, M.-H.; Wang, L.; and Gong, B. 2020. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *CVPR*, 7610–7619.

Kang, B.; Xie, S.; Rohrbach, M.; Yan, Z.; Gordo, A.; Feng, J.; and Kalantidis, Y. 2020. Decoupling Representation and Classifier for Long-Tailed Recognition. In *ICLR*.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Le, Y.; and Yang, X. 2015. Tiny ImageNet Visual Recognition Challenge.

Liang, S.; Li, Y.; and Srikant, R. 2018. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *ICLR*.

Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020. Energy-based Out-of-distribution Detection. In *NeurIPS*.

Liu, Z.; Miao, Z.; Zhan, X.; Wang, J.; Gong, B.; and Yu, S. X. 2019. Large-scale long-tailed recognition in an open world. In *CVPR*, 2537–2546.

Loshchilov, I.; and Hutter, F. 2016. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Meinke, A.; and Hein, M. 2020. Towards neural networks that provably know when they don't know. In *ICLR*.

Menon, A. K.; Jayasumana, S.; Rawat, A. S.; Jain, H.; Veit, A.; and Kumar, S. 2021. Long-tail learning via logit adjustment. In *ICLR*.

Mohseni, S.; Pitale, M.; Yadawa, J.; and Wang, Z. 2020. Self-supervised learning for generalizable out-of-distribution detection. In *AAAI*, 5216–5223.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading digits in natural images with unsupervised feature learning.

Park, S.; Hong, Y.; Heo, B.; Yun, S.; and Choi, J. Y. 2022. The Majority Can Help The Minority: Context-rich Minority Oversampling for Long-tailed Classification. In *CVPR*.

Ren, J.; Yu, C.; sheng, s.; Ma, X.; Zhao, H.; Yi, S.; and Li, h. 2020. Balanced Meta-Softmax for Long-Tailed Visual Recognition. In *NeurIPS*, 4175–4186.

Roy, A. G.; Ren, J.; Azizi, S.; Loh, A.; Natarajan, V.; Mustafa, B.; Pawlowski, N.; Freyberg, J.; Liu, Y.; Beaver, Z.; et al. 2022. Does your dermatology classifier know what it doesn't know? Detecting the long-tail of unseen conditions. *Medical Image Analysis*, 75: 102274.

Tang, K.; Huang, J.; and Zhang, H. 2020. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *NeurIPS*, 1513–1524.

Torralba, A.; Fergus, R.; and Freeman, W. T. 2008. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE TPAMI*, 30(11): 1958–1970.

Van Horn, G.; and Perona, P. 2017. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*.

Vaze, S.; Han, K.; Vedaldi, A.; and Zisserman, A. 2022. Open-Set Recognition: a Good Closed-Set Classifier is All You Need? In *ICLR*.

Vernekar, S.; Gaurav, A.; Abdelzad, V.; Denouden, T.; Salay, R.; and Czarnecki, K. 2019. Out-of-distribution detection in classifiers via generation. *arXiv preprint arXiv:1910.04241*.

Wang, H.; Zhang, A.; Zhu, Y.; Zheng, S.; Li, M.; Smola, A. J.; and Wang, Z. 2022. Partial and Asymmetric Contrastive Learning for Out-of-Distribution Detection in Long-Tailed Recognition. In *ICML*, 23446–23458.

Wang, X.; Lian, L.; Miao, Z.; Liu, Z.; and Yu, S. 2021. Long-tailed Recognition by Routing Diverse Distribution-Aware Experts. In *ICLR*.

Wei, H.; Tao, L.; Xie, R.; and An, B. 2021. Open-set Label Noise Can Improve Robustness Against Inherent Label Noise. In *NeurIPS*.

Wei, H.; Tao, L.; Xie, R.; Feng, L.; and An, B. 2022. Open-Sampling: Exploring Out-of-Distribution data for Re-balancing Long-tailed datasets. In *ICML*, 23615–23630.

Yang, J.; Wang, H.; Feng, L.; Yan, X.; Zheng, H.; Zhang, W.; and Liu, Z. 2021. Semantically coherent out-of-distribution detection. In *ICCV*, 8301–8309.

Yang, Y.; and Xu, Z. 2020. Rethinking the Value of Labels for Improving Class-Imbalanced Learning. In *NeurIPS*.

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J. 2015. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*.

Yun, S.; Han, D.; Chun, S.; Oh, S. J.; Yoo, Y.; and Choe, J. 2019. CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. In *ICCV*, 6022–6031.

Zhong, Z.; Cui, J.; Liu, S.; and Jia, J. 2021. Improving Calibration for Long-Tailed Recognition. In *CVPR*, 16489–16498.

Zhou, B.; Cui, Q.; Wei, X.-S.; and Chen, Z.-M. 2020. BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *CVPR*, 9719–9728.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 40(6): 1452–1464.

Proof of Proposition 1

Proposition 2. *For the cross-entropy loss, Eq. (3) induces gradient noise* $\boldsymbol{g} = -\frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}$ *on* $\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{z}, y)$*, s.t.,* $\boldsymbol{g} \in \mathbb{R}^p, j = \arg\max_{j \in [C+1, C+k]} \widetilde{\boldsymbol{z}}$*. While each OOD sample in OE (Hendrycks, Mazeika, and Dietterich 2019) induces gradient noise* $\boldsymbol{g}' = -\frac{1}{C} \sum_{j=1}^{C} \frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}$ *on* $\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{z}, y)$*, where* $\frac{\cdot}{\widetilde{\boldsymbol{z}}_j}$ *denotes the element-wise division.*

Proof. For Eq. (1), the total gradient is as follows:

$$\begin{aligned} \widetilde{\nabla}_{\boldsymbol{\theta}} \ell_{\text{total}} &= \nabla_{\boldsymbol{\theta}} \ell\left(\boldsymbol{z}, y\right) + \nabla_{\boldsymbol{\theta}} \ell\left(\widetilde{\boldsymbol{z}}, \widetilde{y}\right) \\ &= \nabla_{\boldsymbol{\theta}} \ell\left(\boldsymbol{z}, y\right) + \nabla_{\widetilde{\boldsymbol{z}}} \ell\left(\widetilde{\boldsymbol{z}}, \widetilde{y}\right) \cdot \nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}} \end{aligned}$$

We omit the trade-off parameter λ in Eq. (1) for simplicity. Note that cross-entropy loss is $\ell\left(\boldsymbol{z}, y\right) = -\boldsymbol{e}^{y} \log \boldsymbol{z}$, then the noise imposed on the in-distribution classification loss $\nabla_{\boldsymbol{\theta}} \ell\left(\boldsymbol{z}, y\right)$ is:

$$\begin{aligned} \boldsymbol{g} = \nabla_{\widetilde{\boldsymbol{z}}} \ell\left(\widetilde{\boldsymbol{z}}, \widetilde{y}\right) \cdot \nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}} &= -\left(\frac{e^{\widetilde{y}}}{\widetilde{\boldsymbol{z}}}\right)^{T} \cdot \nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}} \\ = -\sum_{j=1}^{C+k} \left(e_j^{\widetilde{y}} \cdot \frac{\nabla_{\boldsymbol{\theta}_i} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}\right), &\quad \text{s.t. } \widetilde{y} = \arg \max_{j \in [C+1, C+k]} \widetilde{\boldsymbol{z}}. \end{aligned}$$

Since $\boldsymbol{e}^{\widetilde{y}} = (0, \cdots, 1, \cdots, 0)$ is the one-hot vector and only the y-th entry is 1, the expression of the noise $\boldsymbol{z}$ can be simplified as:

$$\boldsymbol{g} = -\frac{\nabla_{\boldsymbol{\theta}} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}, \quad \text{s.t. } j = \arg \max_{j \in [C+1, C+k]} \widetilde{\boldsymbol{z}}.$$

Let g_i be the i-th entry of $\boldsymbol{g}$, we have

$$g_i = -\frac{\nabla_{\boldsymbol{\theta}_i} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j}, \quad \text{s.t. } j = \arg \max_{j \in [C+1, C+k]} \widetilde{\boldsymbol{z}}.$$

While the regularization item of the OE method is $\ell_{\text{OE}} = -\frac{1}{C} \cdot \sum_{i=1}^{C} \log \widetilde{\boldsymbol{z}}_i$.

Then the gradient of ℓ_{OE} w.r.t θ is as follows:

$$\begin{aligned} \nabla_{\theta} \ell_{\text{OE}} = \nabla_{\widetilde{\boldsymbol{z}}} \ell_{\text{OE}} \cdot \nabla_{\theta} \widetilde{\boldsymbol{z}} &= \nabla_{\widetilde{\boldsymbol{z}}} \left(-\frac{1}{C} \cdot \sum_{j=1}^{C} \log \widetilde{\boldsymbol{z}}_j\right) \cdot \nabla_{\theta} \widetilde{\boldsymbol{z}} \\ &= -\frac{1}{C} \cdot \sum_{j=1}^{C} \frac{\nabla_{\theta} \widetilde{\boldsymbol{z}}_j}{\widetilde{\boldsymbol{z}}_j} \end{aligned}$$

□

Additional Experimental Results

On imbalance ratio ρ We use imbalance ratio $\rho = 100$ on both CIFAR10-LT and CIFAR100-LT in routine long-tail OOD experiments. In this section, we show that our method can work well under different imbalance ratios. Specifically, we conduct experiments on CIFAR10-LT with $\rho = 50$. The results are shown in Table 7. Our approach also outperforms the OE by a considerable margin when $\rho = 50$ and outperforms previous state-of-the-art PASCL in 3 out of 4 cases.

Table 7: Results on CIFAR10-LT ($\rho = 50$) using ResNet18.

$\mathcal{D}_{\text{out}}^{\text{test}}$	**Method**	**AUROC** (↑)	**AUPR** (↑)	**FPR95** (↓)	**ACC** (↑)
Average	OE	93.13	91.06	24.73	83.34
	PASCL	93.94	92.79	**22.80**	85.44
	Ours	**94.06**	**93.50**	24.03	**85.57**

On model structures In routine experiments, we use the standard ResNet18 as the backbone model. In this section, we show that our method can work well under different model structures, but conduct experiments using the standard ResNet34 (He et al. 2016). The results are shown in Table 8. Our method consistently outperforms OE and PASCL. In particular, it improves the in-distribution accuracy by 6% in comparison with PASCL.

Table 8: Results on CIFAR10-LT ($\rho = 100$) using ResNet34.

$\mathcal{D}_{\text{out}}^{\text{test}}$	**Method**	**AUROC** (↑)	**AUPR** (↑)	**FPR95** (↓)	**ACC** (↑)
Average	OE	89.86	87.28	33.66	73.39
	PASCL	91.11	89.28	33.21	75.34
	Ours	**93.38**	**92.85**	**26.56**	**82.20**

How does the number of abstention classes affect the performance? Figure 3 shows how the performance changes with different numbers of abstention classes on CIFAR10-LT. Overall the performance is not sensitive in the range chosen. We empirically find that setting $k = 3$ achieves relatively better results with respect to three out of four representative OOD detection performance measures, i.e., AUROC, AUPR, and FPR95.

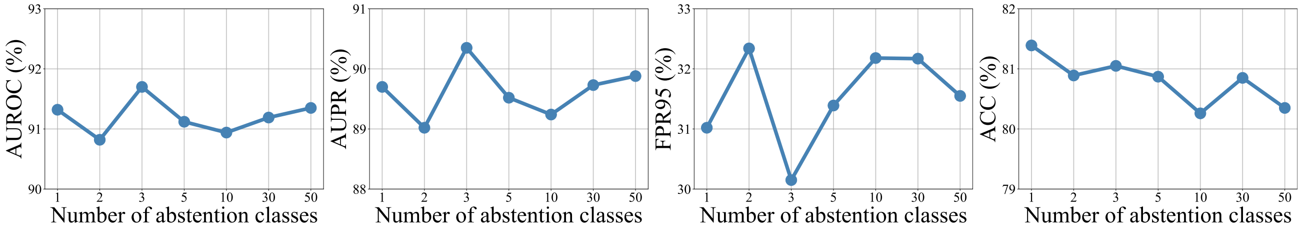

Figure 3: Effect of the number of abstention classes on the model performance.

More visualization Figure 5 illustrates the distribution of OOD scores on the other three OOD datasets considered in this paper. Overall our model distinguishes OOD data from in-distribution data with a clear decision boundary.

Convergence and instability of our method. We present the training loss curve and test accuracy in Figure 4, providing evidence of the stability of our method throughout training. This visualization serves to assure the reviewers that overfitting is not observed in our approach.

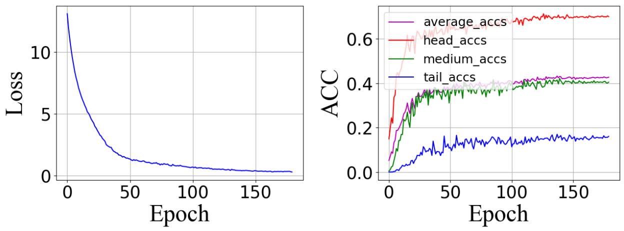

Figure 4: Training loss curve and test accuracy.

Contrast with SOFL

Similar to our approach, SOFL (Mohseni et al. 2020) also uses multiple abstention classes for OOD detection and fine-tunes the model in the second stage. Specifically, SOFL trains on in-distribution data using the cross-entropy loss in the first stage, then uses both in- and out-of-distribution data in the second stage to fine-tune the whole model.

However, as demonstrated in Algorithm 1, we train the model in the first stage using in- and out-of-distribution data. In the second stage, we only fine-tune the final classification layer of the model using in-distribution data and the loss ℓ_{LA} specially designed for long-tailed learning.

Our approach surpasses SOFL in terms of in-distribution classification and OOD detection by a large margin. It is not just about other aspects of our model, but more intuitively, we have superior designs in the two training stages. We use more data in the first stage to make the representation learning of the model more discriminative. The logit adjustment loss adopted in the second stage further boosts the accuracy for tail classes. Moreover, only updating the final classification layer also preserves the model's judgment ability for OOD.

Table 9: We compare the ACC95 of PASCL and ours. Since the number of $\mathcal{D}_{\text{in}}^{\text{test}}$ is $10,000$, we round the product of these numbers to represent the number of correctly classified in-distribution samples in the remaining in-distribution samples when 95% percent of the OOD samples are selected. ACC95 and 1-FPR95 are shown in percentages.

$\mathcal{D}_{\text{out}}^{\text{test}}$	**Method**	**ACC95 (↑)**	**1 - FPR95 (↑)**	N_{correct} (↑)
Texture	OE	71.43	31.72	2266
	PASCL	73.11	32.57	2381
	Ours	73.76	32.47	2395
SVHN	OE	64.27	41.96	2697
	PASCL	64.50	46.55	3002
	Ours	61.67	52.22	3220
CIFAR10	OE	82.67	19.36	1601
	PASCL	82.30	20.45	1683
	Ours	82.61	22.03	1820
Tiny ImageNet	OE	76.22	23.34	1779
	PASCL	77.56	23.89	1853
	Ours	77.07	25.11	1935
LSUN	OE	65.64	36.02	2364
	PASCL	68.05	36.69	2497
	Ours	62.07	44.98	2791
Places365	OE	67.04	34.28	2298
	PASCL	69.04	35.19	2430
	Ours	66.15	39.15	2590
Average	OE	71.21	31.11	2215
	PASCL	72.43	32.56	2358
	Ours	70.55	36.00	2540

Balance of FPR95 and ACC95

Mohseni et al. (Mohseni et al. 2020) show that it is very challenging to achieve high FPR95 and ACC95 concurrently in the OOD detection task. The reason is simple: when FPR95 is low, more in-distribution samples are correctly detected, including those problematic and corner-case in-distribution samples. As a result, the remaining in-distribution samples might incur more misclassification, leading to low ACC95.

In this paper, we raise a question about ACC95: as an indicator of OOD detection, whether a larger ACC95 is preferable when n (in percentage) OOD samples have been successfully detected. Because when a fixed rate of OOD samples is filtered out, the number of the remaining in-distribution samples is not stable among different algorithms or experiment settings. When this number does not fluctuate much among various algorithms, a higher ACC95 indicates better performance. However, there is an alternative interpretation when the number of remaining inliers fluctuates significantly. We compare the results with the previous long-tail OOD detection method PASCL (Wang et al. 2022), and we find that

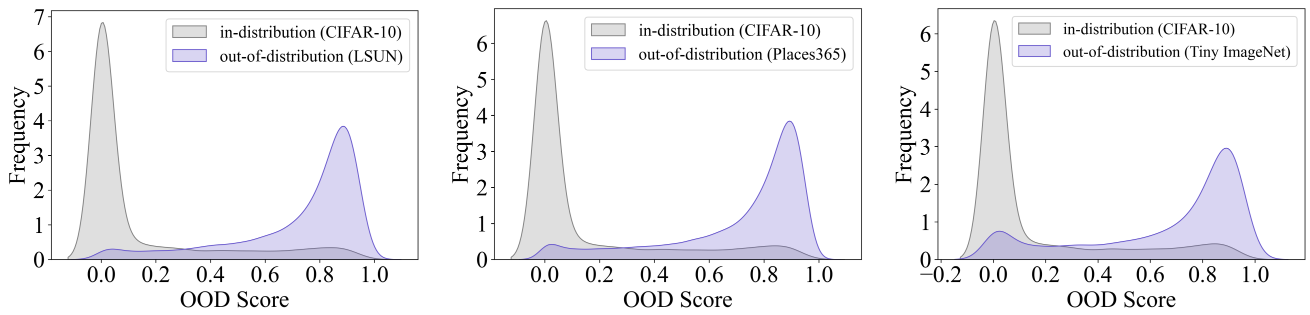

Figure 5: Distribution of OOD scores from our model. The CIFAR10 is used as the in-distribution dataset, and the other six are OOD datasets. It shows that both in-distribution data and OOD data naturally form smooth distributions.

although our approach performs worse than PASCL concerning ACC95 on most of the OOD datasets, the number of remaining in-distribution samples is far more than PASCL. We believe that with the current experiment setups, it makes more sense to compare the number of remaining inlier samples correctly classified after filtering out n (in percentage) OOD samples. The detailed results are shown in table 9.

Therefore, we may combine the evaluation of these two indicators. Assuming the number of $\mathcal{D}_{\text{in}}^{\text{test}}$ is N, the number of correctly classified inlier samples remaining when 95% percent of the OOD samples are filtered out is N_{correct}. Then we have the following formula:

$$N_{\text{correct}} = N \times (1 - FPR95) \times ACC95 \tag{8}$$

In this way, we may evaluate ACC95 and FPR95 in the meantime. It is important to note that N_{correct} has a practical physical significance. When 95% of the OOD samples are filtered out, the remaining correctly classified in-distribution samples. Considering the influence of N on this detection value, N can be omitted when comparing different experiments, and the product of ACC95 and (1-FPR95) can be directly calculated as a measurement. In fact, our method optimizes both ACC95 and FPR95 on CIFAR10-LT. And even though ACC95 performs slightly worse on CIFAR100-LT, the number of the remaining in-distribution samples N_{correct} still outperforms OE and PASCL on each $\mathcal{D}_{\text{out}}^{\text{test}}$ and overall. In addition, the calculation process of N_{correct} involves both the inlier accuracy of the model and the sensitivity to OOD detection, which may make N_{correct} become a new paradigm with both in-distribution classification and OOD detection.

Limitations

While our method examines tail-class augmentation via CutMix, which uses head-class and OOD images as background and tail-class images as the foreground, it results in more reasonable and effective representation learning for tail samples, greatly improving tail part in-distribution accuracy and OOD detection performance. However, since CutMix generates more data, it necessitates additional GPU RAM, albeit with a continuous rise in size. We attempted to minimize extra time overhead and GPU memory consumption by appropriate code architecture and algorithm structure, but we must agree that some extra overhead is unavoidable.

Additionally, this paper focuses on improving the performance by directly using CutMix, that is, we do not adjust the sampling area distribution or fine-tune any other component parts to make this technique more suitable for long-tailed OOD tasks. And we do not rule out the possibility of other more advanced methods achieving this function and yielding better results in the future.

In many real-world applications such as autonomous driving, medical diagnosis, and healthcare, beyond being naturally imbalanced. Moreover, the model usually encounters unknown (out-of-distribution) test data points once deployed. In this paper, we focus on standard classification and out-of-distribution detection as our measure and largely ignore other ethical issues in imbalanced data, especially in minor classes. For example, the data may impose additional constraints on the learning process and final models, e.g., being fair or private. As such, the risk of producing unfair or biased outputs reminds us to carry rigorous validations in critical, high-stakes applications.