Background Data Resampling for Outlier-Aware Classification

Yi Li Nuno Vasconcelos
University of California, San Diego
{yil898,nvasconcelos}@ucsd.edu

Abstract

The problem of learning an image classifier that allows detection of out-of-distribution (OOD) examples, with the help of auxiliary background datasets, is studied. While training with background has been shown to improve OOD detection performance, the optimal choice of such dataset remains an open question, and challenges of data imbalance and computational complexity make it a potentially inefficient or even impractical solution. Targeted at balancing between efficiency and detection quality, a dataset resampling approach is proposed for obtaining a compact yet representative set of background data points. The resampling algorithm takes inspiration from prior work on hard negative mining, performing an iterative adversarial weighting on the background examples and using the learned weights to obtain the subset of desired size. Experiments on different datasets, model architectures and training strategies validate the universal effectiveness and efficiency of adversarially resampled background data. Code is available at https://github.com/JerryYLi/bg-resample-ood.

1. Introduction

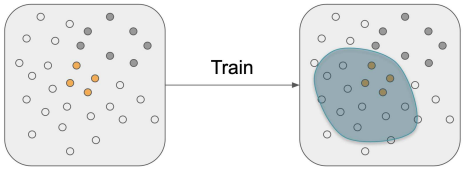
While modern deep neural networks (DNN) achieve or surpass human-level accuracy on image recognition tasks, they are also notorious for producing overconfident decisions on misclassified examples [11, 32], or even inputs that do not belong to any training class [26, 2]. This is problematic for many applications where a) inputs may come from a different distribution than the training data, and b) reliability of prediction is an important concern. Ideally, DNNs should be able to discriminate "outliers" from regular test data (from the training distribution), *i.e*. to detect out-of-distribution (OOD) examples [13].

Recently, various approaches have been proposed to address OOD detection in the context of DNNs that output class probabilities from a softmax layer. Most of this work focuses on improved training through input preprocessing and/or additional loss functions [19, 24, 31, 6, 34]. A less explored alternative is to introduce auxiliary *background* data, sampled outside the training set, for which the classifier is forced to produce low-confidence outputs [22, 7, 14]. This has been proved effective, substantially improving OOD detection quality with no training enhancements other than application of a simple uniformity loss to the background data. On the other hand, a large background dataset, often tens of times the size of the in-distribution (ID) training set, is required. This implies non-trivial increases in storage space and time complexity.

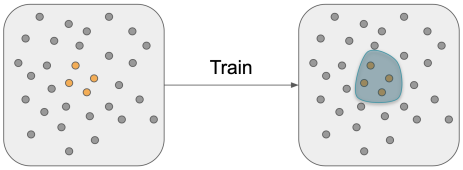
In this work, we consider the problem of optimally compressing a background dataset for OOD purposes. The goal is to, starting from a large pool of background data, identify a compact subset of similar OOD detection performance, *i.e*. such that a model trained on the subset has identical OOD performance to one trained on all the data. The tradeoffs involved in the selection of a good background dataset are illustrated in Figure 1, where orange points represent the ID dataset, open circles the pool of background data, and gray examples the selected subset of OOD examples (OOD dataset). Also shown as a shaded area is the decision rule implemented by the optimal classifier for discrimination of ID vs. OOD data (dark for ID, light for OOD).

When the OOD dataset is small, as in Figure 1a, training is efficient but leads to an inaccurate classifier, since the OOD dataset only covers a small region of background space. High classifier accuracy can be achieved with a very large OOD dataset, as shown in Figure 1b, but this inefficient in computation and memory. A final possibility is to start from the large pool of background data and sample a subset of examples. The simplest form of sampling, illustrated in Figure 1c, is to choose samples independently, using a uniform distribution over the background pool. This is likely better than the approach of (b) but still suboptimal in terms of classifier accuracy.

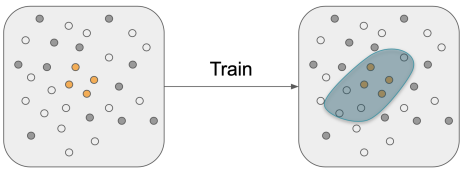
In this work, we seek to develop a sampling strategy that achieves the optimal trade-off between efficiency and OOD detection accuracy. For this, we draw inspiration from hard negative mining in the object detection literature [8, 10], treating OOD detection as a binary classification problem with extremely imbalanced positive (ID) *vs*. negative (OOD) classes. In particular, we propose a dataset resampling scheme that aims to selecting challenging back-



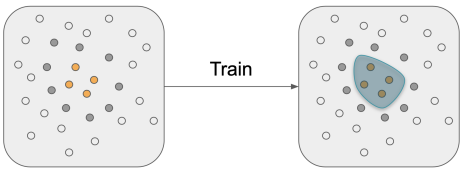
(a) Small background dataset: Efficient but inaccurate.



(b) Large background dataset: Accurate but inefficient.



(c) Uniformly resampled dataset: Efficient but inaccurate.



(d) Adversarially resampled dataset: Accurate and efficient.

Figure 1: **(a)–(c)**: Conventional approach for training outlier-aware classifiers. The performance grows with the size of background data, but as does computational complexity and storage requirement. **(d)**: Proposed dataset resampling scheme, which achieves both accuracy and efficiency. Orange points represent in-distribution data, gray ones are background examples; shaded area denote the decision boundary of trained OOD detector (within which the model predicts in-distribution).

ground images, which are frequently misclassified as in-distribution. As shown in Figure 1d, these are likely to be examples in the ID *vs*. OOD border. The proposed resampling is based on the assignment of a resampling score to each background example, derived from an adversarial reweighting objective that gives higher priority to hard negatives. Resampling scores are then determined by a new adversarial algorithm that minimizes this objective by iterating between two gradient descent steps: 1) Classifier update given reweighted data and 2) weight updates given the new classifier. The learned weights are finally used to determine sampling probabilities to perform example selection.

It is shown that training on the obtained subset of background data leads to similar or higher OOD detection accuracy than using the full background data, while significantly reducing storage space needed per training episode. Experiments also confirm that the proposed adversarial resampling finds datasets of better trade-off between detection quality and training efficiency than uniform example subsampling. This is observed consistently across scenarios with different model architectures, training pipelines and ID datasets.

2. Related Work

Self Awareness. A number of active research areas have focused on the design of self-aware networks. These are networks that "know when they don't know." Self-awareness includes open set recognition, which adds *unknown* classes to a traditional classification problem [2]; confidence calibration, which matches the network output with the true likelihood [11]; and—to be studied in this work—out-of-distribution (OOD) detection, which aims to identify test inputs that come from a distribution different from that seen at training time.

Out-of-distribution detection. The first procedure for OOD detection on deep network classifiers was presented in [13], using the maximum softmax score as an indicator of the likelihood that the input image comes from the same distribution as the training set. Without additional training, the softmax score proved effective for simple in-distribution data (like MNIST [21]) and trivial out-of-distribution examples (like uniform noise). However, the detection quality is far from ideal for more complicated data.

Follow-up work has targeted to improve OOD detection performance by various training enhancements, including input perturbations [19, 24], temperature scaling [24], and network ensembles [19, 31]. Another line of approaches uses *background* examples that do not belong to the training set, as surrogates for the unknown OOD examples at test time. In this case, the classifier is trained with the additional objective of producing uniform (hence low-confidence) softmax scores for background inputs. Background examples can be obtained from either an auxiliary dataset [7, 14] or using a generative model [22]. In particular, [14] showed that large-scale datasets, like Tiny-Images [29] and ImageNet [27], are surprisingly effective as background data, enabling classifiers to learn to discriminate OOD inputs from in-distribution ones.

Training with background data. The use of background data for training has been a standard practice in the design of object detectors, which recognize image patches as foreground (positive) or background (negative) [8, 10, 9]. Due to the imbalance between positive and negative classes, the selection of background patches is crucial for high detection accuracy, and the technique of *hard negative mining* is commonly used to prioritize background examples that are misclassified as foreground [5, 8].

Dataset resampling. Dataset resampling is the technique of undersampling or upsampling examples in a dataset. It is

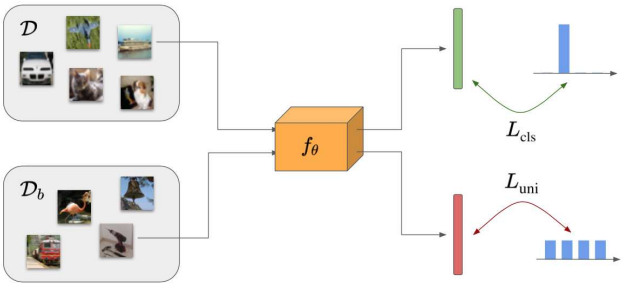


Figure 2: Training pipeline with background examples.

traditionally used to combat class imbalance [3], but can be extended to achieve a wide range of goals. One of the compelling applications is *compressing* a dataset by discarding examples with minimal influence on the performance of trained models [20, 30]. Example selection is also useful for speeding up training by prioritizing informative samples [16], improving test accuracy by discarding examples with noisy labels [1], or removing bias from the data [23].

3. Background Data for OOD Detection

In this section, we discuss the effect of auxiliary background data on the capacity of a trained model to detect out-of-distribution text examples, and motivate the idea of resampling the background dataset.

OOD Formulation. Following [7, 14] we assume a training set-up where data batches are sampled from two datasets, the in-distribution training set $\mathcal{D}$ and background data $\mathcal{D}_b$ (alternatively denoted by outlier exposure set $\mathcal{D}_{\text{OE}}$ in [14]). Classifier θ is then trained to meet two objectives: maximize classification performance on $\mathcal{D}$, while preventing overconfident predictions on $\mathcal{D}_b$. This involves a trade-off as illustrated in Figure 2. While the first goal requires very confident predictions (posterior distributions of low entropy) for in-distribution data, the second requires predictions of very low confidence (high entropy distributions) for OOD data.

This is captured by the objective function

$$L(\theta; \mathcal{D}, \mathcal{D}_b) = L_{\text{in}}(\theta; \mathcal{D}) + \alpha L_{\text{out}}(\theta; \mathcal{D}_b), \tag{1}$$

where

$$L_{\text{in}}(\theta; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(x,y)\in\mathcal{D}} L_{\text{cls}}(f(x;\theta), y) \tag{2}$$

is a classification loss for in-distribution examples, and

$$L_{\text{out}}(\theta; \mathcal{D}_b) = \frac{1}{|\mathcal{D}_b|} \sum_{(x,y)\in\mathcal{D}_b} L_{\text{uni}}(f(x;\theta)) \tag{3}$$

a loss that penalizes high-confidence class predictions on background examples. The hyperparameter α controls the trade-off between the two objectives.

Losses. Unless otherwise noted, we use the standard cross-entropy loss for in-distribution examples

$$L_{\text{cls}}(f(x;\theta), y) = -\log f_y(x;\theta), \tag{4}$$

and the Kullback-Leibler divergence to a uniform class posterior distribution for out-of-distribution inputs

$$L_{\text{uni}}(f(x;\theta)) = -\frac{1}{K}\sum_{k=1}^{K} \log f_k(x;\theta) - \log K. \tag{5}$$

Probabilistic interpretation. Assuming that datasets $\mathcal{D}$ and $\mathcal{D}_b$ are sampled from the task distribution $p_{X,Y}(x,y)$ and background distribution $q_X(x)$ respectively, (1) can be interpreted as an empirical estimate of

$$\begin{aligned} L(\theta; p, q) = \mathbb{E}_{X,Y\sim p(\cdot,\cdot)}[L_{\text{cls}}(f(X;\theta);Y)] \\ + \alpha \mathbb{E}_{X\sim q(\cdot)}[L_{\text{uni}}(f(X;\theta))]. \end{aligned} \tag{6}$$

For the specific losses of (4) and (5), the optimal classifier under (1) is (see supplementary material for derivation)

$$f_k^*(x) = c(x) p_{Y|X}(k \mid x) + \frac{1 - c(x)}{K}, \tag{7}$$

a smoothed version of class probabilities $p_{Y|X}$ by averaging towards uniformity. The smoothing degree is controlled by

$$c(x) = \frac{p_X(x)}{p_X(x) + \alpha q_X(x)}. \tag{8}$$

This can be interpreted as the posterior probability that x is sampled from the task distribution p, given the prior belief that the ratio of in-distribution to background data is $1 : \alpha$. Hence, OOD detection reduces to the binary problem of learning $c(x)$. The k-way probability distribution $p_{Y|X}(k \mid x)$ in (7) is learned by standard classification algorithms.

Generalization behavior. The procedure above has been shown unreasonably effective for OOD detection [22, 7, 14]. Models trained to produce low confidence class predictions on training background data $\mathcal{D}_b$, generalize well to OOD test data $\mathcal{D}_o$, even when $\mathcal{D}_b$ and $\mathcal{D}_o$ come from vastly different domains (*e.g.* natural images in $\mathcal{D}_b$ and noise in $\mathcal{D}_o$). While this generalization ability is not fully understood, empirical studies have shown that a *diverse* set of training background data is important for good test-time performance [14]. On the other hand, it has been shown that *proximity* between $\mathcal{D}_b$ and $\mathcal{D}$ is critical as well [22, 7]. This poses a challenge, since it is usually difficult to find a background dataset that is simultaneously diverse and close to $\mathcal{D}$. In this work, we propose to achieve this goal by selecting examples from large-scale datasets.

4. Background Data Resampling

In this section, we introduce an objective for the optimal resampling of background data for OOD detection, and present a solution based on adversarial reweighting.

4.1. Resampling Objective

Motivated by the observation that training OOD detectors with a very large background dataset $\mathcal{D}_b$ is effective yet inefficient, we propose to sample a subset of examples $\mathcal{D}'_b$ from $\mathcal{D}_b$, that simultaneously satisfies

1. **Efficiency**: Total number of selected examples should not exceed a percentage $\gamma \in (0, 1)$ of the original dataset size, *i.e.*

$$|\mathcal{D}'_b| \leq \gamma |\mathcal{D}_b|. \tag{9}$$

   γ is denoted as the *sample rate*.

2. **Effectiveness**: The estimate of the optimal classifier parameters under objective (1) produced with subset $\mathcal{D}'_b$ and denoted

$$\theta^*(\mathcal{D}'_b) = \arg\min_\theta L(\theta; \mathcal{D}, \mathcal{D}'_b), \tag{10}$$

   should be as close as possible to that obtained from all background examples,

$$\theta^* = \arg\min_\theta L(\theta; \mathcal{D}, \mathcal{D}_b). \tag{11}$$

   The effectiveness of $\mathcal{D}'_b$ is defined as

$$\mathcal{E}(\mathcal{D}'_b) = \frac{L(\theta^*; \mathcal{D}, \mathcal{D}_b)}{L(\theta^*(\mathcal{D}'_b); \mathcal{D}, \mathcal{D}_b)}. \tag{12}$$

Since $\mathcal{E}(\mathcal{D}'_b)$ only depends on D'_b through $L(\theta^*(\mathcal{D}'_b); \mathcal{D}, \mathcal{D}_b)$, it is equivalent to optimize the former or the latter. Hence, the two goals above can be met by solving the following constrained optimization problem

$$\begin{aligned} \mathcal{D}^*_b = \arg\min_{\mathcal{D}'_b \subseteq \mathcal{D}_b} \quad & \mathcal{F}\left[L(\theta^*(\mathcal{D}'_b); \mathcal{D}, \mathcal{D}_b)\right], \\ \text{subject to} \quad & |\mathcal{D}'_b| \leq \gamma |\mathcal{D}_b|. \end{aligned} \tag{13}$$

where $\mathcal{F}$ is a function discussed in Section 4.3. This, however, is a combinatorial problem whose complexity grows rapidly with the size of background dataset $|\mathcal{D}_b|$, making it impractical to find the exact solution. We next propose an alternative solution based on learning to reweight examples.

4.2. Example Reweighting

Since the resampling objective of (13) is combinatorial, we seek a differentiable relaxation based on a set of continuous example weights. Formally, we assign to each example $x_i \in \mathcal{D}_b$ a weight $w_i \geq 0$. By interpreting this weight as the relative frequency of x_i in the resampled subset $\mathcal{D}'_b$, the OOD detection loss after reweighting can be written as

$$\begin{aligned} L_{\text{out}}(\theta; w) &= \frac{1}{|\mathcal{D}'_b|} \sum_{(x,y) \in \mathcal{D}'_b} L_{\text{uni}}(f(x; \theta)) \\ &= \frac{1}{\sum_i w_i} \sum_{i=1}^{|\mathcal{D}_b|} w_i L_{\text{uni}}(f(x_i; \theta)). \end{aligned} \tag{14}$$

The optimal parameter set of (10) is then

$$\theta^*(w) = \arg\min_\theta L(\theta; \mathcal{D}, w) \tag{15}$$

$$= \arg\min_\theta L_{\text{in}}(\theta; \mathcal{D}) + \alpha L_{\text{out}}(\theta; w) \tag{16}$$

and the optimization of (13) becomes

$$w^* = \arg\min_w \mathcal{F}\left[L(\theta^*(w); \mathcal{D}, \mathcal{D}_b)\right], \tag{17}$$

under the size constraint that we leave for later discussion in Section 4.3. This problem can be solved by alternatingly optimizing for w and $\theta^*(w)$, *i.e.* iterating between (16) and the solution of (17) given $\theta^*(w)$:

$$\theta^{(t)} = \arg\min_\theta \left[L_{\text{in}}(\theta; \mathcal{D}) + \alpha L_{\text{out}}(\theta; w^{(t-1)})\right] \tag{18}$$

$$w^{(t)} = \arg\min_w \mathcal{F}\left[L_{\text{in}}(\theta^{(t)}; \mathcal{D}) + \alpha L_{\text{out}}(\theta^{(t)}, w)\right] \tag{19}$$

The *parameter update* step of (18) consists of the design of a classifier given the reweighted dataset, using a combination of the cross entropy loss of (4) and the OOD detection loss of (14), and solved by backpropagation. Given suitable $\mathcal{F}$, the *weight update* step of (19) is a continuous function of w and can also be solved by backpropagation.

4.3. Adversarial Resampling

A natural choice for $\mathcal{F}$ is the identity. In this case, (17) is equivalent to maximizing the effectiveness $\mathcal{E}(\mathcal{D}'_b)$ of the resampled dataset, given in (12). Hence, the steps of (18) and (19) *collaborate* to find the most effective background dataset. While intuitive, our experience is that this solution is too greedy and converges to poor local minima in a few iterations. To see this, assume that the parameter update step produced a solution $\theta^{(t)}$. The weight update step then seeks to minimize $L_{\text{out}}(\theta^{(t)}, w)$. Under the constraint of sampling rate γ, the optimal solution to this problem is to assign all weight mass to the $\gamma|\mathcal{D}_b|$ examples x_i of lowest $L_{\text{uni}}(f(x_i, \theta))$, *i.e.* the examples of most uniform posterior distribution under the classifier of parameters $\theta^{(t)}$. These are likely to be the examples x_i *farthest away* from the region of support of the ID dataset $\mathcal{D}$. At step $t+1$ they are unlikely to have large effect on the optimization of (18), because they already have a small OOD loss $L_{\text{out}}(\theta; w^{(t-1)})$. Hence, there is little incentive for $\theta^{(t+1)}$ to differ much from $\theta^{(t)}$ and the optimization converges in a few iterations. In summary, the problem is that the collaborative nature of the two steps does not force the optimization to *explore* the space of background datasets, or even select background examples that overlap with the ID dataset.

This observation motivated us to consider an alternative *adversarial* sampling strategy, where the weight update step attempts to *minimize* the efficiency of the background dataset. This can be easily enforced by selecting

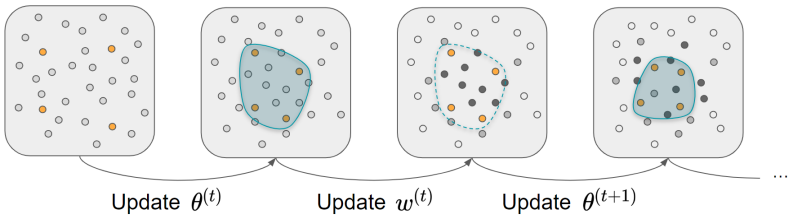


Figure 3: Graphical illustration of proposed adversarial resampling procedure. Following Fig. 1, orange pts denote ID examples, gray ones are background data, with darker shades representing higher resampling weights.

Algorithm 1: Adversarial resampling, batch version.

Input: ID dataset $\mathcal{D}$, background dataset $\mathcal{D}_b$, pre-trained classifier θ, learning rate η_θ, η_w, loss coefficient α, total iterations T

Initialize: $w^{(0)} \leftarrow [1, \ldots, 1]$, $\theta^{(0)} \leftarrow \theta$;
for $t = 0, \ldots, T-1$ **do**
  Compute ID loss $l_{\text{in}}^{(t)} \leftarrow L_{\text{in}}(\theta^{(t)}; \mathcal{D})$;
  Compute OOD loss $l_{\text{out}}^{(t)} \leftarrow L_{\text{out}}(\theta^{(t)}; w^{(t)})$;
  Update classifier
    $\theta^{(t+1)} \leftarrow \theta^{(t)} - \eta_\theta \nabla_{\theta^{(t)}} \left(l_{\text{in}}^{(t)} + \alpha l_{\text{out}}^{(t)} \right)$;
  Update weights
    $w^{(t+1)} \leftarrow w^{(t)} + \eta_w \nabla_{w^{(t)}} l_{\text{out}}^{(t)}$;

Output: Resampling weights $w^{(T)}$.

$\mathcal{F}[L] = -L$, leading to the procedure of Algorithm 1 (based on *batch* gradient descent; see supplementary for practical SGD optimization). In this case, as shown in Figure 3, given $\theta^{(t)}$, the optimal solution of (19) is to assign most weight to the examples of *largest* OOD loss. These are the examples that have the least entropic posterior distribution and are most likely to be close to the ID dataset $\mathcal{D}$ or even overlap it. Hence, at step $t+1$, there is a strong incentive to modify the parameters of the classifier, so as to minimize the OOD loss component of (18). In result, the optimization is forced to explore the space of background datasets, choosing a background dataset of significant example diversity and examples on the boundary between the ID data $\mathcal{D}$ and the background data. It should be noted that this behavior is similar to that of hard negative mining techniques used to tackle the imbalance between positive and negative examples in object detection [5, 8, 10, 9].

It is also important to note that, under the adversarial strategy, there are no trivial solutions to (19), and the reweighting can be computed independently of the target sampling rate γ. Once the optimal resampling weights $\{w_i\}_{i=1}^{|\mathcal{D}_b|}$ are found, the resampled dataset $\mathcal{D}_b'$ s obtained by selecting each example x_i independently with probability

$$p_i = \min\left(1, \frac{\gamma |\mathcal{D}_b|}{\sum_j w_j} w_i\right), \tag{20}$$

leading to an expected dataset size of $\mathbb{E}[|\mathcal{D}_b'|] = \sum_i p_i \leq \gamma |\mathcal{D}_b|$ that satisfies the efficiency constraint of (9).

In-dist.	# train / # test	Ref.
CIFAR-10 [18]	50,000 / 10,000	[13, 24, 22, 31, 6, 7, 14]
CIFAR-100 [18]	50,000 / 10,000	[13, 24, 31, 14]
Tiny ImageNet[1]	100,000 / 10,000	[14]
Out-of-dist.	**# test**	**Ref.**
Gaussian	–	[13, 24, 22, 31, 6, 14]
Uniform	–	[13, 24, 31, 6]
Textures [4]	5,640	[14]
LSUN [33]	10,000	[24, 22, 31, 6, 14]
SVHN [25]	26,032	[22, 7, 14]
Places [36]	328,500	[14]

Table 1: In-distribution and out-of-distribution datasets for experimental evaluation. Most are common choices in prior work.

5. Experiments

In this section we present an experimental evaluation of the proposed dataset resampling method.

5.1. Experimental Setup

Datasets. The OOD data for test-time evaluation is a pool of datasets that do not overlap the in-distribution data used to train the classifier. Since no universal protocol exists for selecting the training dataset and OOD test sets, we use the combination of noise and natural image datasets summarized in Table 1. As shown in the table, most of these datasets have been used in previous works.

Among the works that used background data for training, there is also no agreement upon the selection of background dataset $\mathcal{D}_b$: When training a CIFAR-10 classifier, [22] used SVHN as $\mathcal{D}_b$, while [7] used CIFAR-100, and [14] Tiny Images. We chose to instead use the ILSVRC'12 dataset [27]. This was mostly for its diversity, making the background dataset a better representative of unseen OOD data. We show in Section 5.2 that using ILSVRC as background data does indeed enable superior detection performance on test-time OOD datasets.

Models. We use a 40-layer Wide Residual Network (WRN) [35], in alignment with previous work on OOD detection [13, 24, 31, 6]. The model is pre-trained on the in-distribution dataset $\mathcal{D}$ for 100 epochs, and fine-tuned on both $\mathcal{D}$ and the background data $\mathcal{D}_b$ for another 50 epochs, using the loss of (1). The initial learning rate is set to 0.1 for pre-training and 0.001 for fine-tuning, and is reduced by 10 times every 30 epochs. Like [14], we use $\alpha = 0.5$ to

[1] https://tiny-imagenet.herokuapp.com/

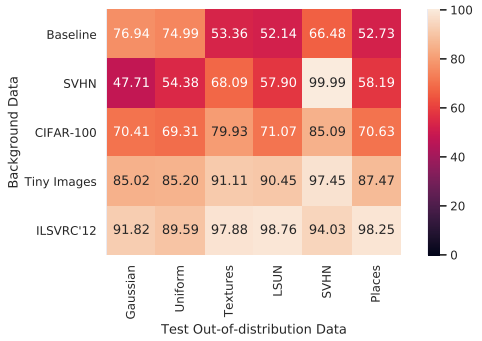


Figure 4: Detection AUPR% using different background datasets. In-distribution dataset is CIFAR-10.

balance classification cross-entropy loss L_{in} and OOD detection loss L_{out}. This ensures good separation of ID and OOD examples, without significantly affecting the classification accuracy on the test set (by $\sim 1\%$).

Criteria. Following [13, 24, 14] we use the maximum score at the output of the softmax layer of classifier to decide on ID *vs*. OOD and report the detection performance, measured using three different criteria:

- **FPR95**: Detection false positive rate at 95% true positive rate, *i.e*. the proportion of ID data misclassified as OOD, for detection threshold such that 95% of the OOD examples are detected. *Lower is better.* Note that we are treating OOD data as *positive* here.
- **AUROC**: Area under ROC curve, which plots the true positive rate against false positive rate as detection threshold increases from 0 to 1. *Higher is better.*
- **AUPR**: Area under precision-recall (PR) curve. *Higher is better.* Also known as average precision, we use AUPR in alignment with AUROC metric.

The sizes of OOD datasets differ greatly, creating a variable ratio between positive and negative classes. This makes the AUPR metric not directly comparable across datasets (whereas FPR95 and AUROC remain relatively invariant). To compensate for this we follow [14], which randomly downsamples all OOD datasets to 20% of the ID dataset size, with 5 repetitions per dataset, and report the average OOD detection performance. Since the standard deviation of most measurements is small, we leave it out of the main text; see supplementary material for more details.

5.2. OOD Detection with Background Data

We start by investigating the effect of background datasets on test time OOD detection accuracy. Using CIFAR-10 as ID dataset $\mathcal{D}$, we consider four choices of background data $\mathcal{D}_b$: SVHN, CIFAR-100, Tiny Images, and ILSVRC'12. We expect larger datasets to lead to better results, as they are more diverse and likely to cover the wide spectrum of data unseen in $\mathcal{D}$. Figure 4 compares the trained models in terms of OOD AUPR. Several conclusions can be drawn. First, all models trained with background data improve over the baseline (no background data) for at least one of the background datasets. Second, Tiny Images and ILSVRC'12 perform the best. This confirms the hypothesis that large-scale background datasets improve OOD detection. Third, the classifier trained with ILSVRC'12 background data performed the best in 5 of 6 test sets, achieving an average AUPR of 95.06%. For this reason, ILSVRC is used as source of background data in the remaining experiments. It should be noted, however, that the performance gains cannot be explained uniquely by dataset size. For example, the model trained on 80 million Tiny Images has an average AUPR of 89.45%. This is lower than that of the model trained on the 1.28 million examples of ILSVRC'12.

Background $\mathcal{D}_b$	FPR95 ↓	AUROC ↑	AUPR ↑
None [13], $\gamma = 0$	31.45	90.72	62.77
Full, $\gamma = 100\%$	2.21	**99.41**	**95.06**
Random, $\gamma = 10\%$	2.85	99.14	92.92
Resampled, $\gamma = 10\%$	**1.94**	99.37	94.16

(a) In-distribution $\mathcal{D}$ = CIFAR-10.

Background $\mathcal{D}_b$	FPR95 ↓	AUROC ↑	AUPR ↑
None [13], $\gamma = 0$	54.81	76.71	33.98
Full, $\gamma = 100\%$	8.51	97.03	81.16
Random, $\gamma = 10\%$	11.08	96.08	76.17
Resampled, $\gamma = 10\%$	**6.40**	**97.76**	**83.75**

(b) In-distribution $\mathcal{D}$ = CIFAR-100.

Background $\mathcal{D}_b$	FPR95 ↓	AUROC ↑	AUPR ↑
None [13], $\gamma = 0$	62.41	72.01	30.73
Full, $\gamma = 100\%$	3.77	99.39	97.70
Random, $\gamma = 10\%$	8.17	98.19	95.22
Resampled, $\gamma = 10\%$	**1.25**	**99.64**	**98.86**

(c) In-distribution $\mathcal{D}$ = Tiny ImageNet.

Table 2: OOD detection performance (in %) on CIFAR-10, CIFAR-100 and Tiny ImageNet, using different background data for training. Results are averaged over 6 test OOD sets in Table 1; see supp. material for individual measures.

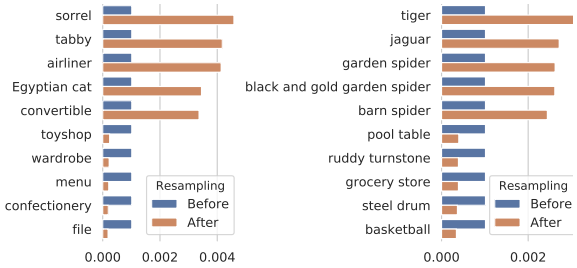
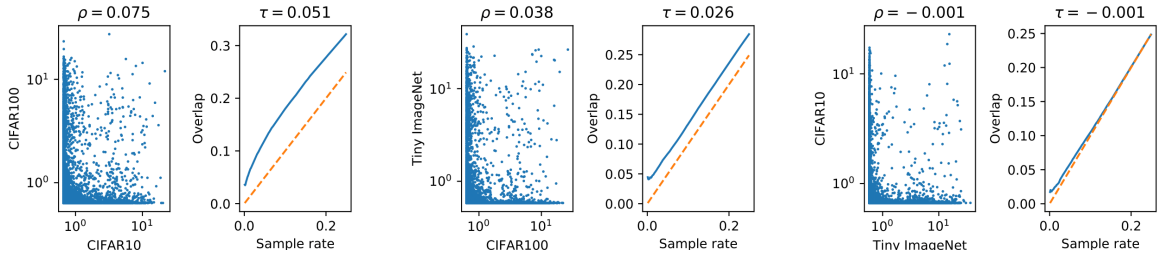


Figure 5: 5 most & least frequent background classes after resampling for CIFAR-10 (**left**) and CIFAR-100 (**right**).

5.3. Background Data Resampling

While large-scale background datasets like ILSVRC'12 improve OOD detection, they require a non-trivial increase in storage space relative to the ID training set. We next evaluate the OOD detection of convnets trained on different ID datasets. In all cases background examples are ILSVRC images, but selected with different approaches:



(a) CIFAR-10 *vs*. CIFAR-100. (b) CIFAR-100 *vs*. Tiny ImageNet. (c) Tiny ImageNet *vs*. CIFAR-10.

Figure 6: Correlation plots of ISLVRC resampling weights for different ID dataset pairs (**left**) and overlap between resampled data (**right**).

- **No** background data: The standard cross-entropy classifier baseline.
- **Full** background data: Optimizes the OOD detection loss of (3) on all examples of $\mathcal{D}_b$. This requires the most additional space & time complexity.
- **Random** selected background data: Optimizes (3) on a random subset of 10% examples from $\mathcal{D}_b$.
- **Resampled** background data: Optimizes (3) on the subset of examples from $\mathcal{D}_b$ produced by the proposed dataset resampling algorithm.

For a fair comparison, we perform example selection using sampling rate $\gamma = 0.1$; the effect of varying γ is discussed in section 5.4. See supplementary material for further implementation details of the resampling.

Detection quality. Table 2 summarizes the results averaged over all OOD datasets (see supplementary for breakdown by test set). Confirming the observations of [14], all methods that use background data substantially outperform the standard cross-entropy classifier. When background data is used, uniformly subsampling degrades the OOD detection accuracy of using full $\mathcal{D}_b$. The proposed resampling method, however, does not suffer from the same performance loss. Notably, on CIFAR-100 and Tiny ImageNet, models trained with 10% background data even outperform their counterparts trained on the full background. This is likely due to the emphasis of resampling on examples close to the in-distribution, forcing the network to learn a more precise decision boundary.

Resampled data. Figure 5 shows the background classes most upsampled and downsampled through the resampling process. It can be observed that the proposed algorithm has a clear preference towards classes semantically close to the ID dataset: Of the five most frequent classes in the resampled background data for CIFAR-10, *tabby/Egyptian cat*, *airliner* and *convertible* are closely related to ID classes *cat*, *airplane* and *automobile* respectively. This makes intuitive sense as the model trained on CIFAR-10 is likely to produce high confidence outputs for these images, failing to discriminate them from ID data.

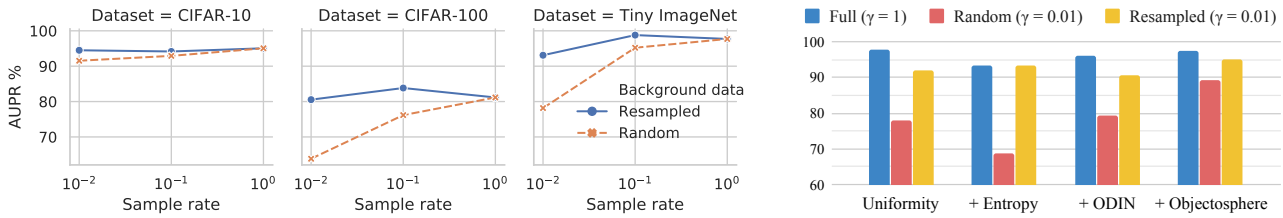
Figure 6 shows the scatter plots for the weights learned for pairs of ID datasets, as well as their rank correlation coefficients [28, 17]. We also visualize the ratio of overlap between the resampled datasets using both sets of weights as the sampling rate γ varies and compare it to chance level. A large weight correlation implies that the optimal background datasets for the two training sets share more examples in common. This is a desirable property, as the resampled dataset learned for one in-distribution task could be used to train other datasets. Indeed, it can be observed that the examples learned for CIFAR-100 were positively correlated with those for CIFAR-10 and Tiny ImageNet. We will see in Section 5.5 that these examples do generalize as background data across tasks.

5.4. Training on Resampled Datasets

Sampling rate. Having seen that it is possible to drastically reduce the size of background data while maintaining the OOD detection accuracy, we further reduce the sampling rate to $\gamma = 0.01$ to investigate the effectiveness of the proposed approach under conditions where the storage budget is very limited. Figure 7a illustrates the detection performance as a function of the size of background dataset. The models trained using both proposed and randomly resampled data saw a drop in OOD detection performance as γ is further decreased, yet the advantage of adversarial resampling over random selection remains significant.

Auxiliary OOD training. We note that all experiments above have used the KL divergence to uniform distribution on the background data as training-time OOD detection loss (5), the standard approach adopted in [22, 7, 14]. Also canonically, OOD detection at test-time is performed by thresholding the maximum softmax scores [13]. We now evaluate the compatibility of the proposed resampled background datasets with alternative methods commonly used for out-of-distribution detection:

- **Entropy** maximization replaces the uniformity loss of (5) by the negative entropy of posterior probabilities predicted by the classifier, given by $L_{\text{ent}}(f(x;\theta)) = \sum_{k=1}^{K} f_k(x;\theta) \log f_k(x;\theta)$.
- **ODIN** [24] uses two simple techniques at training time to calibrate classifier predictions, namely temperature-scaled softmax $s_k(T) = \frac{e^{v_k/T}}{\sum_{j=1}^{K} e^{v_j/T}}$, and input perturbation $\hat{x} = x + \epsilon \operatorname{sign}(\nabla_x \log \max_k s_k(T))$.



(a) Detection AUPR% *vs*. sampling rate γ.

(b) Detection AUPR% with auxiliary methods.

Figure 7: Training with resampled background datasets.

Background $\mathcal{D}_b$	Architectures			
	WRN40	WRN28	Res18	Dense100
Random, $\gamma = 10\%$	76.17	86.71	74.38	74.72
Transferred, $\gamma = 10\%$	–	**87.33**	**81.48**	**76.50**
Native, $\gamma = 10\%$	**83.75**	86.05	80.75	74.95
Random, $\gamma = 1\%$	63.84	81.53	74.46	66.96
Transferred, $\gamma = 1\%$	–	**85.88**	76.97	75.81
Native, $\gamma = 1\%$	**80.54**	85.68	**78.28**	**76.93**

(a) AUPR% of new model architectures trained on CIFAR-100. All **transferred** background data are resampled for WRN40; while **native** ones are resampled for their respective architectures.

Background $\mathcal{D}_b$	In-distribution $\mathcal{D}$		
	CIFAR100	CIFAR10	TinyImgNet
Random, $\gamma = 10\%$	76.17	92.92	95.22
Transferred, $\gamma = 10\%$	–	**94.56**	97.45
Native, $\gamma = 10\%$	**83.75**	94.16	**98.79**
Random, $\gamma = 1\%$	63.84	91.54	83.68
Transferred, $\gamma = 1\%$	–	94.12	**94.82**
Native, $\gamma = 1\%$	**80.54**	**94.50**	93.10

(b) AUPR% of WRN-40 models trained on new datasets. All **transferred** background data are resampled for CIFAR-100; while **native** ones are resampled for their respective ID datasets.

Table 3: Generalization capacity of resampled data across models (**left**) and in-distribution datasets (**right**).

- **Objectosphere loss** [7] aims at minimizing the feature magnitude of background examples, which naturally results in a uniform classifier output when the classification layer has no bias term.

Figure 7b shows the OOD detection performance when the model is trained and/or tested using the above approaches, again on full, random and resampled background data learned earlier. The resampling method provides consistent improvement over random sampling, proving to be a reliable complementary to previously proposed algorithms.

5.5. Generalization in Retraining

One of the greatest advantage of having a compact and representative set of background examples is that the storage space and time complexity are greatly reduced. This is especially relevant in the scenario where models are retrained multiple times, either with different architectures or on a different dataset. Therefore, it would be desirable that the resampled dataset remains effective when model architectures, training procedure, and ID datasets change. In the following experiments we re-evaluate the OOD detection quality of models under these changes, as a measure of generalization capacity of the resampled dataset.

Across model architectures. We start by considering whether the weights learned with a WRN-40 [35] classifier are effective for convnets with other architectures. Table 3a shows the OOD detection performance of these weights for three alternative architectures: WRN-28, DenseNet-100 [15], and ResNet-18 [12]. The table shows that there is a noticeable advantage of using resampled datasets over random selection, even when retraining different networks. In many instances, using the transferred background data from WRN-40 even provides similar performance to when using the optimal resampled dataset for the new model.

Across in-distribution datasets. The correlation plots of Figure 6 suggest that resampled dataset for one ID dataset may be helpful for training new datasets. Table 3b shows the OOD detection performance of retrained models on CIFAR-10 and Tiny ImageNet, using the resampled background data learned for CIFAR-100. Again, the weights learned by the proposed algorithm demonstrated its robustness across tasks, yielding comparable OOD detection performance to native resampling. We find this result rather inspiring, as it shows the potential of building a universal background set of examples that can be used to augment an arbitrary dataset to make it *compatible* for OOD detection.

6. Conclusion

We presented a resampling approach to select informative background examples from large-scale datasets for out-of-distribution detection. Motivated by hard negative mining in object detection, we developed an adversarial optimization procedure that learns a set of weights for selecting challenging background examples. Using a small sampling rate, we were able to obtain compact resampled datasets that are often as effective as using full background data, sometimes even improving OOD detection quality. The resampling method was shown to work well in conjunction with auxiliary training algorithms in the literature, and generalizable across models and in-distribution tasks.

Acknowledgement This work was partially funded by NSF awards IIS-1637941, IIS-1924937, and NVIDIA GPU donations.

References

[1] Anelia Angelova, Yaser Abu-Mostafam, and Pietro Perona. Pruning training sets for learning of object categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 494–501. IEEE, 2005.

[2] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1563–1572, 2016.

[3] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *Journal of Artificial Intelligence Research (JAIR)*, 16:321–357, 2002.

[4] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3606–3613, 2014.

[5] N Dalal and B Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893. IEEE, 2005.

[6] Terrance DeVries and Graham W Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018.

[7] Akshay Raj Dhamija, Manuel Günther, and Terrance Boult. Reducing network agnostophobia. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 9157–9168, 2018.

[8] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(9):1627–1645, 2009.

[9] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[10] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587, 2014.

[11] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 1321–1330. JMLR. org, 2017.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[13] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[14] Dan Hendrycks, Mantas Mazeika, and Thomas G Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations (ICLR)*, 2019.

[15] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.

[16] Angelos Katharopoulos and Francois Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International Conference on Machine Learning (ICML)*, pages 2530–2539, 2018.

[17] Maurice George Kendall. Rank correlation methods. 1948.

[18] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[19] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6402–6413, 2017.

[20] Agata Lapedriza, Hamed Pirsiavash, Zoya Bylinskii, and Antonio Torralba. Are all training examples equally valuable? *arXiv preprint arXiv:1311.6510*, 2013.

[21] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[22] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations (ICLR)*, 2018.

[23] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9572–9581, 2019.

[24] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[25] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[26] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 427–436, 2015.

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[28] C Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 1904.

[29] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern*

Analysis and Machine Intelligence (TPAMI), 30(11):1958–1970, 2008.

[30] Kailas Vodrahalli, Ke Li, and Jitendra Malik. Are all training examples created equal? an empirical study. *arXiv preprint arXiv:1811.12569*, 2018.

[31] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 550–564, 2018.

[32] Pei Wang and Nuno Vasconcelos. Towards realistic predictors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 36–51, 2018.

[33] 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. *arXiv preprint arXiv:1506.03365*, 2015.

[34] Qing Yu and Kiyoharu Aizawa. Unsupervised out-of-distribution detection by maximum classifier discrepancy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9518–9526, 2019.

[35] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[36] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(6):1452–1464, 2017.