SSB: Simple but Strong Baseline for Boosting Performance of Open-Set Semi-Supervised Learning

Yue Fan Anna Kukleva Dengxin Dai Bernt Schiele
{yfan, akukleva, ddai, schiele}@mpi-inf.mpg.de
Max Planck Institute for Informatics, Saarbrücken, Germany
Saarland Informatics Campus

Abstract

Semi-supervised learning (SSL) methods effectively leverage unlabeled data to improve model generalization. However, SSL models often underperform in open-set scenarios, where unlabeled data contain outliers from novel categories that do not appear in the labeled set. In this paper, we study the challenging and realistic open-set SSL setting, where the goal is to both correctly classify inliers and to detect outliers. Intuitively, the inlier classifier should be trained on inlier data only. However, we find that inlier classification performance can be largely improved by incorporating high-confidence pseudo-labeled data, regardless of whether they are inliers or outliers. Also, we propose to utilize non-linear transformations to separate the features used for inlier classification and outlier detection in the multi-task learning framework, preventing adverse effects between them. Additionally, we introduce pseudo-negative mining, which further boosts outlier detection performance. The three ingredients lead to what we call ***S****imple but* ***S****trong* ***B****aseline (SSB) for open-set SSL. In experiments, SSB greatly improves both inlier classification and outlier detection performance, outperforming existing methods by a large margin. Our code will be released at `https://github.com/YUE-FAN/SSB`.*

1. Introduction

Semi-supervised learning (SSL) has achieved great success in improving model performance by leveraging unlabeled data [26, 25, 42, 29, 4, 3, 41, 45, 11, 50, 44]. However, standard SSL assumes that the unlabeled samples come from the same set of categories as the labeled samples, which makes them struggle in open-set settings [33], where unlabeled data contain out-of-distribution (OOD) samples from novel classes that do not appear in the labeled set (see Fig. 1). In this paper, we study this more realistic setting called *open-set semi-supervised learning*, where the goal is to learn both a good closed-set classifier to classify inliers and to detect outliers as shown in Fig. 1.

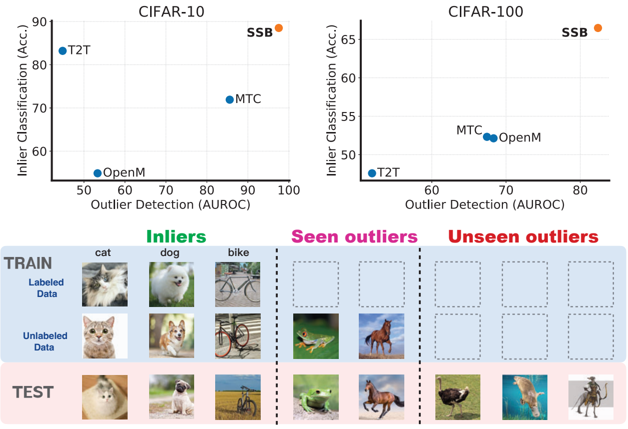


Figure 1: Open-set semi-supervised learning considers a realistic and challenging setting, where unlabeled data contains samples from novel classes (**seen outliers**) that do not appear in the labeled data. At test time, the model should correctly classify **inliers**, while identifying outliers seen during the training and, most importantly, **unseen outliers** that do not appear in the training set. We measure test accuracy for the inlier classification performance and AUROC for the outlier detection performance. Our method (SSB) achieves superior performance in both tasks.

Recent works on open-set SSL [20, 7, 38, 48, 14, 16, 17, 21] have achieved strong performance [43, 28, 19, 1] through a multi-task learning framework, which consists of an inlier classifier, an outlier detector, and a shared feature encoder, as shown in Figure 2. The outlier detector is trained to filter out OOD data from the unlabeled data so that the classifier is only trained on inliers. However, this framework has two major drawbacks. First, detector-based filtering often removes many inliers along with OOD data, leading to suboptimal classification performance due to the low utilization ratio of unlabeled data. Second, the inlier classifier which shares the same feature encoder with the outlier detector can have an adverse effect on the detection performance as shown in Table 1.

To this end, we contribute a **S**imple but **S**trong **B**aseline, **SSB**, for open-set SSL with three ingredients to address the above issues. (1) In contrast to detector-based filtering aiming to remove OOD data, we propose to incorporate pseudo-labels with high inlier classifier confidence into the training, *irrespective of whether a sample is an inlier or OOD*. This not only effectively improves the unlabeled data utilization ratio but also includes many useful OOD data that can be seen as natural data augmentations of inliers (see Fig. 5). (2) Instead of directly sharing features between the classifier and detector, we add non-linear transformations for the task-specific heads and find that this effectively reduces mutual interference between them, resulting in more specialized features and improved performance for both tasks. (3) In addition, we propose pseudo-negative mining to further improve outlier detector training by enhancing the data diversity of OOD data with pseudo-outliers. Despite its simplicity, SSB achieves significant improvements in both inlier classification and OOD detection. As shown in Fig. 1, existing methods either struggle in detecting outliers or have difficulties with inlier classification while SSB obtains good performance for both tasks.

2. Related Work

Semi-supervised learning. Semi-supervised learning (SSL) aims to improve model performance by exploiting both labeled and unlabeled data. As one of the most widely used techniques, pseudo-labeling [26] is adopted by many strong SSL methods [41, 4, 4, 45, 50, 2, 35, 46, 5, 27]. The idea is to generate artificial labels for unlabeled data to improve model training. [4, 3] compute soft pseudo-labels and then apply MixUp [51] with labeled data to improve the performance; [41, 45, 50] achieves good performance by combining pseudo-labeling with consistency regularization [25, 29, 42]; [35] proposes a meta learning approach that uses a teacher model to refine the pseudo-labels based on the training of a student model; [46] leverages the idea of self-training which generates pseudo labels in an iterative way and inject noise to each training stage. In this paper, we also adopt a simple confidence-based pseudo-labeling [41] for classifier training, which is an effective way of leveraging unlabeled data to improve the model performance. Compared to standard SSL, SSB has an additional outlier detector, which enables the model to reject samples that do not belong to any of the inlier classes.

Open-set SSL & Class-mismatched SSL. First shown by [33], standard SSL methods suffer from performance degradation when there are out-of-distribution (OOD) samples in unlabeled data. Since then, various approaches have been proposed to address this challenge [7, 14, 48, 38, 20, 34, 16, 17, 21]. Existing methods seek to alleviate the effect of OOD data by filtering them out in different ways so that the classification model is trained with inliers only. For example, [7] uses model ensemble [40] to compute soft pseudo-labels and performs filtering with a confidence threshold; [14] proposes a bi-level optimization to weaken the loss weights for OOD data; [48] assigns an OOD score to each unlabeled data and refines it during the training; [38] leverages one-vs-all (OVA) classifiers [39] for OOD detection and propose a consistency loss to train them; [20] proposes a cross-modal matching module to detector outliers. [21] employs adversarial domain adaptation to filter unlabeled data and find recyclable OOD data to improve the performance; [16] uses energy-discrepancy to identify inliers and outliers. In contrast, we show that if the representations of the inlier classifier and the outlier detector are well-separated, OOD data turns out to be a powerful source to improve the inlier classification without degrading the detection performance. So, instead of filtering OOD data, we use a simple confidence-based pseudo-labeling to incorporate them into the training.

Open-world SSL. Open-set SSL is similar to open-world SSL [6, 36, 37] but bears several important differences. While both have unlabeled data of novel classes during the training, the goal of open-world SSL is to classify inliers and discover new classes from OOD data instead of rejecting them. Another important difference is that open-world SSL is often a transductive learning setting while open-set SSL requires generalization beyond the current distribution. Namely, the model should be able to detect OOD data from novel classes that present in the training set as well as OOD data from classes that are never seen during training.

3. SSB: Simple but Strong Baseline for Open-Set Semi-Supervised Learning

In this section, we first present the problem setup of open-set semi-supervised learning (SSL). Then, we give an overview of our method SSB in Section 3.1 before presenting details of the three simple yet effective ingredients used in our method in Section 3.2, 3.3, and 3.4.

Problem setup and notations: As shown in Fig. 1, open-set SSL generalizes the settings of standard SSL and out-of-distribution (OOD) detection. It considers three disjoint sets of classes: $\mathcal{C}$ corresponds to the inlier classes that are partially annotated, $\mathcal{U}_{\mathcal{S}}$ contains the outlier classes seen during training but without annotations, and lastly, $\mathcal{U}_{\mathcal{U}}$ is composed of the classes that are not seen during training (only seen at test time). The training data contains a small labeled set $\mathcal{D}_{\text{labeled}} = \{(\mathbf{x}_i^l, y_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{C}$ and a large unlabeled set $\mathcal{D}_{\text{unlabeled}} = \{(\mathbf{x}_i^u)\}_{i=1}^M \subset \mathcal{X}$, where $\mathcal{X}$ is the input space. While the labeled set only consists of samples of in-

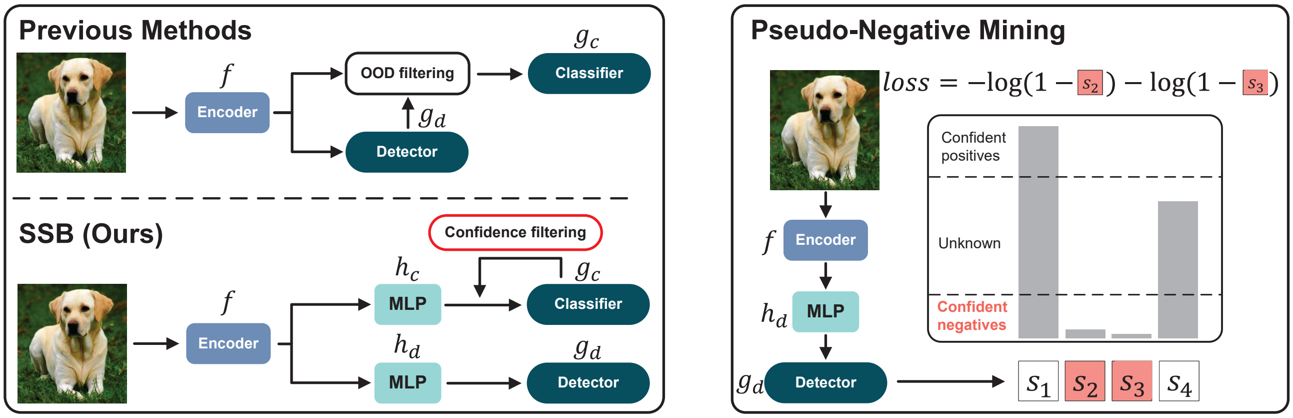


Figure 2: **Left:** Our baseline for open-set SSL consists of an inlier classifier g_c, an outlier detector g_d, and a shared feature encoder f whose features are separated from the task-specific heads by two projection heads h_c and h_d. Unlike the detector-based filtering, we adopt confidence-based pseudo-labeling by the inlier classifier to leverage useful OOD data for classifier training. For detector training, we train one-vs-all (OVA) classifiers as in OpenMatch [38]. **Right:** Given the inlier scores (s_1 to s_4), pseudo-negative mining selects confident negatives (s_2 and s_3 in the figure), whose inlier scores are lower than a pre-defined threshold, as pseudo-outliers to help the outlier detector training.

lier classes, the unlabeled set contains both samples from $\mathcal{C}$ and $\mathcal{U}_\mathcal{S}$. Thus, the the ground-truth label of $\mathbf{x}^u$ is from $\mathcal{C} \cup \mathcal{U}_\mathcal{S}$ with $\mathcal{C} \cap \mathcal{U}_\mathcal{S} = \emptyset$.

The goal of open-set SSL is to train a model that can perform good inlier classification as well as detecting both seen and unseen outliers. Without loss of generality, consider a test set $\mathcal{D}_{\text{test}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N} \subset \mathcal{X} \times (\mathcal{C} \cup \mathcal{U}_\mathcal{S} \cup \mathcal{U}_\mathcal{U})$, where $\mathcal{C} \cap \mathcal{U}_\mathcal{U} = \emptyset$ and $\mathcal{U}_\mathcal{S} \cap \mathcal{U}_\mathcal{U} = \emptyset$. The learned model should be able to correctly classify inliers $\{(\mathbf{x}_i | y_i \in \mathcal{C})\}$ and detect outliers from $\{(\mathbf{x}_i | y_i \in \mathcal{U}_\mathcal{S})\}$ as well as $\{(\mathbf{x}_i | y_i \in \mathcal{U}_\mathcal{U})\}$, which is crucial for practical applications.

3.1. Method Overview

Following [20, 7, 38, 48, 14, 16, 17, 21], we adopt a multi-task learning framework for open-set SSL, which performs inlier classification and outlier detection. As shown in Fig. 2, SSB comprises four components: (1) An inlier classifier g_c, (2) an outlier detector g_d, (3) a shared feature encoder f, and (4) importantly, two projection heads h_c and h_d. Inspired by [38], the outlier detector g_d consists of $|\mathcal{C}|$ one-vs-all (OVA) binary classifiers, each of which is trained to distinguish inliers from outliers for each single class. Given a batch of labeled data $\mathbf{X}^l = \{(\mathbf{x}_i^l, y_i)\}_{i=1}^{B_l}$ and unlabeled data $\mathbf{X}^u = \{(\mathbf{x}_i^u)\}_{i=1}^{B_u}$, the total loss for training the model is:

$$L_{total} = L_{cls}(\mathbf{X}^l, \mathbf{X}^u; f, h_c, g_c) + L_{det}(\mathbf{X}^l, \mathbf{X}^u; f, h_d, g_d) \quad (1)$$

where L_{cls} and L_{det} are the classification and detection losses, respectively. For the sake of brevity, we will drop the dependencies of the loss function on f, h_c, g_c, h_d, and g_d in the following. The complete algorithm of SSB is summarized by Alg. 1 in Appendix D.

During inference, the test image is first fed to the inlier classifier to compute the class prediction. Then, the corresponding detector is used to decide whether it is an inlier of the predicted class or an outlier. We explain the details of SSB in the following three sections.

3.2. Boosting Inlier Classification with Classifier Pseudo-Labeling

Existing methods for open-set SSL [20, 7, 38, 48, 14] aim to eliminate OOD data from the classifier training. This is typically accomplished by training outlier detectors that can filter out OOD data from unlabeled data, as shown in Fig. 2. However, as we will see in Table 3, detector-based filtering often removes many inliers along with OOD data, which leads to a low utilization ratio of unlabeled data and hinders inlier classification performance.

In this work, instead of using detector-based filtering, we propose to incorporate unlabeled data with confident pseudo-labels (as generated by the *inlier classifier*) into the training, *irrespective of whether it is inlier or OOD data.* This not only effectively improves the unlabeled data utilization ratio but also includes many useful OOD data as natural data augmentations of inliers into the training (see Fig. 5). Inspired by [41], we train the model with pseudo-labels from the inlier classifier whose confidence scores are above a pre-defined threshold. Specifically, for each unlabeled sample $\mathbf{x}_i^u$, we first predict the pseudo-label distribution as $\hat{p}_i^u = \text{softmax}(h_c(g_c(f(\mathbf{x}_i^u))))$. Then, the confidence score of the pseudo-label is computed as $\max \hat{p}_i^u$. Finally, the cross-entropy loss is calculated for samples whose pseudo-labels have confidence scores greater than a pre-defined threshold τ as:

$$L^u_{cls}(\mathbf{X}^u) = \frac{1}{B_u}\sum_{i=1}^{B_u} \mathbb{1}(\max \hat{p}^u_i \geq \tau) H(\hat{p}^u_i, \hat{y}^u_i) \tag{2}$$

where $H(\cdot,\cdot)$ denotes the cross-entropy, $\hat{y}^u_i = \text{argmax} \hat{p}^u_i$, and $\mathbb{1}(\cdot)$ is the indicator function which outputs 1 when the confidence score is above the threshold τ.

The total classification loss is computed as the summation of a labeled data loss and the unlabeled data loss as:

$$L_{cls}(\mathbf{X}^l, \mathbf{X}^u) = L^l_{cls}(\mathbf{X}^l) + L^u_{cls}(\mathbf{X}^u) \tag{3}$$

where L^l_{cls} is a standard cross-entropy loss for labeled data.

Despite its simplicity, we obtain a substantial performance improvement in inlier classification through classifier confidence-based pseudo-labeling as shown in Table 1. Our method is conceptually different from previous methods as we aim to leverage OOD data rather than remove them. On the one hand, our method effectively improves the unlabeled data utilization ratio as shown in Table 3, which leads to great inlier classification performance improvement. On the other hand, our method provides an effective way of leveraging useful OOD data for classifier training. In fact, many OOD data are natural data augmentations of inliers and are beneficial for classification performance if used carefully. As shown in Fig. 5, the selected OOD data present large visual similarities with samples of inlier classes, and, thus, significantly enhance the data diversity, leading to improved generalization performance.

3.3. Non-Linear Feature Boosting

In previous methods, simply including OOD samples into the classifier training harms detection performance since the inlier classifier and the outlier detector use the same feature representation [38, 48, 20]. On the one hand, the classifier uses OOD data as pseudo-inliers, thus mixing their representations in the feature space. On the other hand, the outlier detector is trained to distinguish inliers and outliers, which leads to separated representations in the feature space. As a result, the contradiction between the classifier and the outlier detector ultimately adversely affects each other, which limits the overall performance, as shown in Table 1.

In this work, we find empirically that simply adding non-linear transformations between the task-specific heads and the shared feature encoder can effectively mitigate the adverse effect. Given a sample $\mathbf{x}_i$, two multi-layer perceptron (MLP) projection heads h_c and h_d are used to transform the features from the encoder. The output of the network is thus $h_c(g_c(f(\mathbf{x}_i)))$ for the classifier and $h_d(g_d(f(\mathbf{x}_i)))$ for the outlier detector. Compared to the previous methods, the non-linear transformations effectively prevent mutual interference between the classifier and detector, resulting in more specialized features and improved performance in both tasks. In Table 1, while the OOD detection performance degenerates when adding OOD data for classifier training for the model without the projection heads, SSB, in contrast, still exhibits excellent performance in detecting outliers with the help of the projection heads. Moreover, the efficacy of the non-linear projection head also generalizes to other frameworks. We show in the experiment section that it is compatible with various SSL backbones and open-set SSL methods and leads to performance improvement.

3.4. Outlier Detection with Pseudo-Negative Mining

In this section, we first describe the outlier detector used in SSB and then introduce a simple yet effective technique called pseudo-negative mining to improve the outlier detector training.

Following [38], we adopt $|\mathcal{C}|$ one-vs-all (OVA) binary classifiers for OOD detection, where each OVA classifier is trained to distinguish between inliers and outliers for each individual inlier class. Given a labeled sample $\mathbf{x}^l_i$ from class y_i, it is regarded as an inlier for class y_i and an outlier for class $k, k \neq y_i$. Therefore, the OVA classifiers can be trained using binary cross-entropy loss on the positive-negative pairs constructed from the labeled set as:

$$L^l_{det}(\mathbf{X}^l) = -\frac{1}{B_l}\sum_{i=1}^{B_l} log(p_{y_i}(\mathbf{x}^l_i)) + \frac{1}{K}\sum_{k\neq y_i} log(1-p_k(\mathbf{x}^l_i)) \tag{4}$$

where $p_k(\mathbf{x}^l_i)$ is the inlier score of $\mathbf{x}^l_i$ for class k computed by the k-th detector and $K = |\mathcal{C}| - 1$.

However, due to data scarcity, it is difficult to learn good representations for outliers with labeled data only. To this end, we propose pseudo-negative mining to further improve the outlier detector training by leveraging confident negatives as pseudo-outliers to enhance the data diversity of OOD data. As shown in Fig. 2, given an unlabeled sample $\mathbf{x}^u_i$, we consider it as a pseudo-outlier for class k if the inlier score for class k is lower than a pre-defined threshold. Then, $\mathbf{x}^u_i$ is used as a negative sample to calculate the cross-entropy loss of class k. The final loss for $\mathbf{x}^u_i$ is the summation over all classes using it as the negative sample:

$$L^u_{det}(\mathbf{x}^u_i) = -\frac{1}{\sum_k \mathbb{1}(p_k < \theta)}\sum_{k=1}^{|\mathcal{C}|} \mathbb{1}(p_k < \theta) log(1-p_k(\mathbf{x}^u_i)) \tag{5}$$

where p_k is the inlier score from the k-th detector and $\mathbb{1}(\cdot)$ is the indicator function which outputs 1 when the confidence score is less than the threshold θ. This increases the data diversity of outliers and improves generalization performance as shown in Table 5. Compared to standard pseudo-labels, pseudo-outliers have much higher precision because we specify which classes the sample does not belong to rather than which class it belongs to. The latter is a more difficult task than the former. Therefore, pseudo-negative mining is less susceptible to inaccurate predictions while increasing data utilization.

Our final loss for detector training also includes Open-set Consistency (OC) loss [38] and entropy minimization (EM) [12] because they can lead to further improvement. The overall loss for training the detector is as follows:

$$L_{det}(\mathbf{X}^l, \mathbf{X}^u) = L^l_{det}(\mathbf{X}^l) + \lambda^u_{det} L^u_{det}(\mathbf{X}^u) \\ + \lambda^u_{OC} L^u_{OC}(\mathbf{X}^u) + \lambda^u_{em} L^u_{em}(\mathbf{X}^u) \quad (6)$$

where λ^u_{det}, λ^u_{OC}, and λ^u_{em} are loss weights; L^u_{OC} is the soft open-set consistency regularization loss, which enhances the smoothness of the OVA classifier with respect to input transformations; L^u_{em} is the entropy minimization loss, which encourages more confident predictions.

4. Experiments

In this section, we first compare SSB with existing methods in Section 4.1, and then provide an ablation study and further analysis in Section 4.2.

4.1. Main Results

Datasets & Evaluation. As mentioned in Section 3, the goal of open-set SSL is to train a good inlier classifier as well as an outlier detector that can identify both seen and unseen outliers. Therefore, we need to construct three class spaces: inlier classes $\mathcal{C}$, seen outlier classes $\mathcal{U}_\mathcal{S}$, and unseen outlier classes $\mathcal{U}_\mathcal{U}$. For each setting: the labeled set contains samples from $\mathcal{C}$ only; the unlabeled set contains samples from $\mathcal{C}$ and $\mathcal{U}_\mathcal{S}$; the test set contains samples from $\mathcal{C}$, $\mathcal{U}_\mathcal{S}$, and $\mathcal{U}_\mathcal{U}$. The inlier classification performance is evaluated on $\mathcal{C}$ using test accuracy as in standard supervised learning. The OOD detection performance is measured by AUROC following [38] and we report the **average performance in detecting seen outliers and unseen outliers** (see Appendix A for separate AUROC on seen outliers and unseen outliers).

Following [38], we evaluate SSB on CIFAR-10 [24], CIFAR-100 [24], and ImageNet [10] with different numbers of labeled data. For CIFAR-10, the 6 animal classes are used as inlier classes, and the rest 4 are used as seen outlier classes during the training. Additionally, test sets from SVHN [31], CIFAR-100, LSUN [47], and ImageNet are considered as unseen outliers, and used to evaluate the detection performance on unseen outliers. For CIFAR-100, the inlier-outlier split is performed on super classes, and two settings are considered: 80 inlier classes (20 outlier classes) and 55 inlier classes (45 outlier classes). Similar to CIFAR-10, test sets from SVHN, CIFAR-10, LSUN, and ImageNet are used to evaluate the detection performance on unseen outliers. For ImageNet, we follow [38] to use ImageNet-30[18], which is a subset of ImageNet containing 30 distinctive classes. The first 20 classes are used as inlier classes while the rest 10 are used as outlier classes. Stanford Dogs [22], CUB-200 [9], Flowers102 [32], Caltech-256 [13], Describable Textures Dataset [8], LSUN are used as unseen outlier classes at test time.

Implementation details. We use Wide ResNet-28-2 [49] as the backbone for CIFAR experiments and ResNet-18 [15] for ImageNet experiments. As standard SSL models do not have the notion of OOD detection, we adopt the method in [18], where the OOD score of an input image $\mathbf{x}$ is computed as $1 - \max \text{softmax}(f(\mathbf{x}))$ and f denotes the model. Thus, the input image is considered as an outlier if the OOD score is higher than a pre-defined threshold. For other open-set SSL methods, we directly employ the authors' implementations and follow their default hyper-parameters.

For SSB, we use two two-layer MLPs with ReLU [30] non-linearity to separate representations for all settings. The hidden dimension is 1024 for CIFAR settings and 4096 for ImageNet settings. For classifier training, we follow [41] and set the threshold τ as 0.95. For outlier detector training, we set λ^u_{det} as 1 for all settings and follow [38] for the weights of OC loss and entropy minimization. The threshold θ is 0.01 for all experiments (see ablation in Appendix C). Following [38], we train our model for 512 epochs with SGD [23] optimizer. The learning rate is set as 0.03 with a cosine decay. The batch size is 64. Additionally, we defer the training of the outlier detector until epoch 475 to reduce the computational cost as we find empirically the deferred training does not comprise the model performance. The ablation on the deferred training is in Appendix C.

When combined with standard SSL methods (e.g. SSB + FlexMatch), we replace the classifier training losses in Equation 1 with the corresponding losses of different methods while keeping the outlier detector the same. When combined with open-set SSL methods (e.g. MTC + SSB), we make three modifications. First, we separate the outlier detector branch from the classifier branch using the proposed MLP projection head. Second, we replace the outlier detector training losses with our loss from Equation 6. Third, we do not filter unlabeled data with the outlier detector for classifier training.

Results. We compare SSB with both standard SSL and open-set SSL methods. Fig. 3 and 4 summarize the inlier test accuracy and outlier AUROC for CIFAR datasets and ImageNet, respectively. Considering the goal of open-set SSL is to achieve *both good inlier classification accuracy and outlier detection*, SSB greatly outperforms standard SSL methods in outlier detection, and open-set SSL methods in inlier classification. For example, on CIFAR-10 with 25 labels, the AUROC of our best method is 11.97% higher than the best method excluding ours. Moreover, when combined with standard SSL algorithms, our method demonstrates consistent improvement in OOD detection, and in

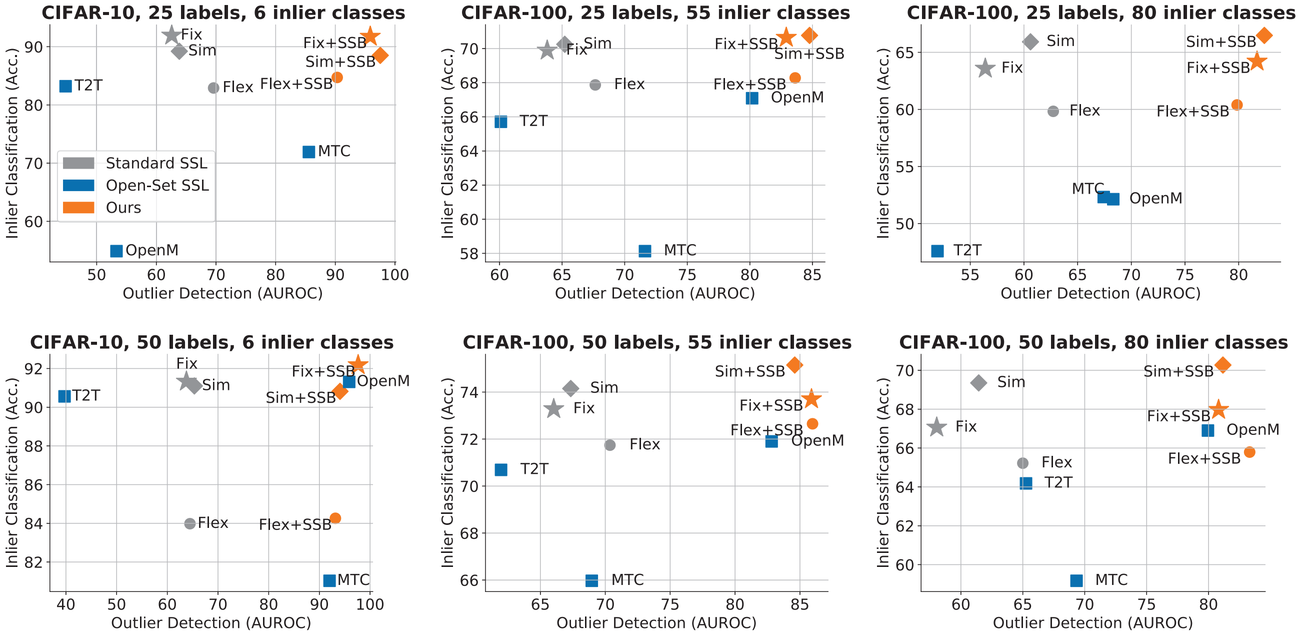


Figure 3: **Classification and detection performance on CIFAR-10 and CIFAR-100 with varying numbers of inlier classes and labeled data.** We measure test accuracy for the inliers classification performance and AUROC for the outlier detection performance. While standard SSL methods suffer in outlier detection and open-set SSL methods suffer in inlier classification, SSB achieves good performance in both tasks. Noted that the reported outlier detection performance is the *average AUROC in detecting both seen and unseen outliers*. Please see Appendix A for a detailed breakdown of the results in tables and results on more benchmarks.

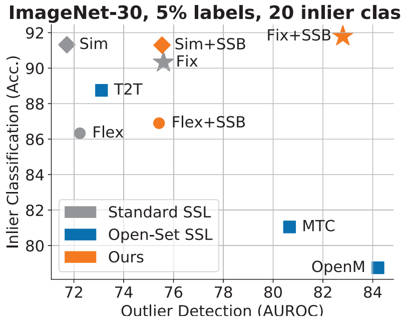


Figure 4: **Classification performance versus the outlier detection performance on ImageNet-30.** SSB achieves good performance in both inlier classification and OOD detection. Please see Appendix A for a detailed breakdown of the results in tables.

most cases, better test accuracy for inlier classification. This suggests the flexibility of our method, which makes it possible to benefit from the most advanced approaches. Note that the performance improvement of SSB can not be simply explained by the increased number of parameters introduced in the projection heads. Please see Fig. 7 for a comparison between SSB and other methods + MLP heads.

Additionally, SSB is more robust to the number of labeled data than others. We achieve reasonable performance given a small number of labeled data while other methods fail to generalize. For example, on CIFAR-10 with 6 inlier classes, OpenMatch has similar inlier accuracy as ours at 50 labels. When the number of labeled data is halved, their performance decreases to 54.88% while our method still has a test accuracy of 91.74%. Please see Appendix B for comparisons on more benchmarks.

4.2. Ablation Study

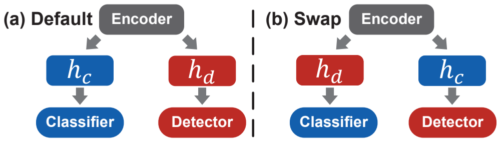
In this section, we analyze the design choices of SSB and show their importance through ablation experiments. If not specified, we use CIFAR-10 with 25 labeled data as our default setting for ablation. The same data split is used for fair comparison.

Importance of non-linear projection heads. As mentioned in Section 3, we use 2-layer MLPs to mitigate the adverse effect between the inlier classifier and outlier detector. Here we study the effect of the projection heads in Table 1. As we can see, incorporating confidence filtering yields a significant improvement in inlier classification performance (resulting in a 12.23% to 13.18% increase). However, the OOD detection performance experiences a substantial decline when the projection heads are missing (AUROC from 89.67% to 63.46%). This is because the classifier tends to mix the features of inliers and outliers with the same pseudo-labels in a shared feature space, which contradicts the goal of the outlier detector. The addition of the projection heads not only restores the OOD detection performance but also achieves superior results when combined with confidence filtering. Adding the projection heads in combination with confidence filtering not only restores the OOD detection performance but achieves even better performance, which indicates the importance of representation separation. Note that it is important to have two independent projection heads for the inlier classifier and outlier detector. A shared projection head does not restore the OOD detection performance as shown in Table 1. Moreover, we show in Table 2 that both classification and detection performance degrade when swapping the task-specific features of a pre-trained model with the fixed encoder. In particular, when re-training the detector (just a fully-connected layer) on top of classification features, seen AUROC drops from 89.18% to 53.99%, which suggests our model learns more task-specific features. Therefore, the utilization of the projection heads separates concerns between the classifier and detector, which eases the difficulties of the task and allows them to be trained jointly without adversely affecting each other. The effect of the depth and the width of the projection head is studied in Appendix C.

Proj. head	Conf. filter	Inlier Cls. (Acc.)	Outlier Det. (AUROC)
		78.05	89.67
shared		76.75	91.92
separate		78.47	90.92
	✓	90.28	63.46
shared	✓	90.93	63.87
separate	✓	**91.65**	**94.76**

Table 1: **Effect of the projection head and confidence-based pseudo-labeling for classifier training.** We use a 2-layer MLP as the projection head. All models are trained with pseudo-negative mining on the same data split.

Improving data utilization with confidence-based pseudo-labeling. Here we study the effect of different classifier training strategies. We compare three unlabeled data filtering methods for classifier training: (1) *det.* selects pseudo-inliers with the outlier detector as in [38]; (2) *det. (tuned)*, where we choose the selection threshold in detector-based filtering so that the recall of actual inlier samples matches ours; (3) *conf.* uses unlabeled data whose confidence is higher than a pre-defined threshold, which is our method. As shown in Table 3, although *det.* successfully removes many OOD data, it also eliminates many inliers, resulting in a low utilization ratio of unlabeled data (0.29% unlabeled data are used in training). In contrast, our method includes pseudo-labels with high classifier confidence into the training, irrespective of whether a sample is out-of-distribution, which leads to a high utilization ratio of unlabeled data (94.22%), thus, outperforming *det.* with a large margin. Moreover, our method also outperforms *det (tuned)* whose data selection threshold is tuned for better performance. This is because we incorporate a significant amount of OOD data in the training process (40.16% v.s. 16.90%). In fact, many OOD data are natural data augmentation of inliers, which can substantially improve closed-set classification if used carefully. When removing pseudo-labeled OOD data using an oracle during the training. The inlier classification accuracy decreases by 3.37% on CIFAR-10 with 25 labels (from 91.65% to 88.28%), which suggests pseudo-labeled OOD data are helpful for inlier classification. In Fig. 5, we visualize top-5 confident OOD samples predicted for three inlier classes from *conf.* on CIFAR-100. We can see that the selected samples are related to the inlier classes and contain the corresponding semantics despite being outliers. For example, OOD data selected for *sea* are images with sea background (more examples in Appendix E).



Nearest Neighbor	Inlier Cls. (Acc.)	Outlier Det. (AUROC)
default (a)	**55.04**	**99.43**
swap cls. & det. features (b)	53.70	77.89

Table 2: **Classification and detection performance using features of different heads.** We fix the encoder and MLP heads and evaluate the classification and detection performance using nearest neighbors on labeled set. Our model learns specialized features since swapping h_c and h_d leads to inferior performance in both tasks.

Effect of pseudo-negative mining. Table 5 shows the effect of pseudo-negative mining. We compare our pseudo-negative mining with standard pseudo-labeling which predicts artificial labels for unlabeled data and uses confident predictions with labeled data loss. While standard pseudo-labeling does not help the OOD detection performance further, pseudo-negative mining improves the seen AUROC by 4.73% over the model without pseudo-negative mining. Compared to standard pseudo-labeling, pseudo-negative mining not only includes more unlabeled data into the training, but also presents high precision for the selected

	Filter method	det.	det. (tuned)	conf. (ours)
Test	Inlier Clf. (Acc.)	47.20	86.53	**91.65**
	Outlier Det. (AUROC)	57.72	87.87	**94.76**
Train	Utilization ratio of:			
	- Unlabeled	0.29	58.09	94.22
	- OOD data	0.04	16.90	40.16
	Prec. of pseudo-inliers	95.17	86.53	58.30
	Recall of inliers	0.47	93.86	92.14

Table 3: **Effect of different OOD filtering methods for classifier training.** We compare three filtering methods: *conf.* denotes the confidence-based pseudo-labeling; *det.* uses the outlier detector to select pseudo-inliers for classifier training; *det. (tuned)* is a tuned version of *det.* that matches the recall of inliers with our method. We compare the performance as well as the data utilization ratio, precision, and recall of the inliers from unlabeled data during training. All models are trained with pseudo-negative mining and the projection head on the same data split.

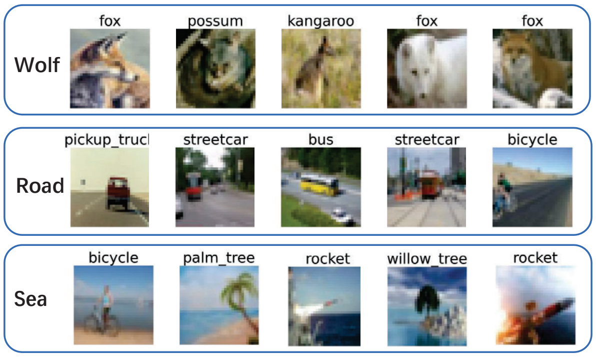


Figure 5: **OOD samples can be used as data augmentation to improve the generalization performance.** The figure shows three semantic classes from labeled data (wolf, road, and sea), and top-5 confident OOD samples predicted for those classes. The ground-truth semantic class of the OOD sample is on the top of each image. We can see that OOD data with high confidence present large visual similarities to the corresponding semantic classes.

pseudo-outliers as shown in Fig. 6.

As mentioned in Section 3.4, we utilize unlabeled data with low inlier scores as pseudo-outliers to enhance the data diversity of outlier classes. An unlabeled sample is used as a pseudo-outlier only if its confidence score is less than a pre-defined threshold θ. Table 4 compares the results of different thresholds. We can see that our method achieves similar performance as long as θ takes a relatively small value, which suggests the good robustness of our method against this hyper-parameter. We provide more ablation on loss weight and data augmentation in Appendix C.

Threshold θ	Inlier Cls. (Acc.)	Outlier Det. (seen AUROC)
0.2	91.87	92.96
0.1	**92.03**	93.16
0.05	91.97	94.21
0.01	91.65	**94.76**
0.005	91.52	94.75
0.001	91.70	94.15

Table 4: **Effect of different thresholds θ for pseudo-negative mining.** Our method shows good robustness against a wide range of thresholds. We use CIFAR-10 with 25 labeled data here.

Pseudo-labeling	Inlier Cls. (Acc.)	Outlier Det. (AUROC)
None	91.52	90.03
Standard	91.63	89.69
Pseudo-neg.	**91.65**	**94.76**

Table 5: **Effect of pseudo-negative mining for OOD detection.** All models are trained with confidence-based pseudo-labeling and a 2-layer MLP projection head on the same data split.

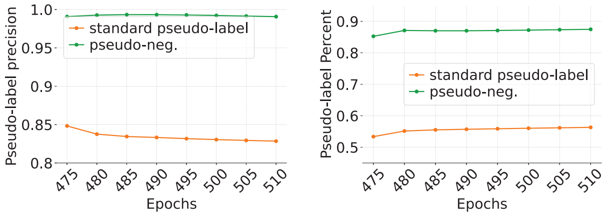


Figure 6: Compared to standard pseudo-labeling, pseudo-negative mining has not only higher prediction precision, but also higher data utilization rate.

Ablation on outlier detectors. Here, we compare the performance of different outlier detection methods. Specifically, we choose three schemes from recent works, including the binary classifier from MTC [48], cross-modal matching from T2T [20], and OVA classifiers from Open-Match [38]. As shown in Table 6. While all methods show reasonable performance, OVA classifiers exhibit the best performance in both inlier classification and OOD detection. Hence, we use OVA classifiers as the outlier detector in our final model.

OOD Detector	Inlier Cls. (Acc.)	Outlier Det. (AUROC)
binary classifier [48]	70.93	76.12
cross-modal matching [20]	69.27	75.99
OVA classifiers [38]	**71.00**	**82.62**

Table 6: **Comparison between different outlier detectors.** The experiment is conducted on CIFAR-100 with 55 inlier classes and 25 labels per class.

Compatibility with other open-set SSL methods. We evaluated the compatibility of our method with other open-set SSL techniques in Table 7. Our results indicate that our method is highly compatible, as all existing methods showed improved performance in both inlier classification and outlier detection when combined with our approach. This demonstrates the flexibility of our method and suggests that it can be easily integrated into existing frameworks as a plug-and-play solution.

	Inlier Cls. (Acc.)	Outlier Det. (AUROC)
MTC	60.24	69.88
MTC + Ours	**60.42**	**74.98**
T2T	64.78	52.93
T2T + Ours	**66.98**	**69.50**
OpenMatch	68.53	80.00
OpenM. + Ours	**71.00**	**82.62**

Table 7: **Integrating our method with other open-set SSL methods improves performance.** The setting is CIFAR-100 with 55 inlier classes and 25 labels per class.

Equal-parameter comparison. As mentioned in Section 4.1, the performance improvement of SSB can not be simply explained by the increased number of parameters introduced in the projection heads. Here we compare SSB with other methods + MLP heads so that they have the same number of parameters as SSB. As shown in Fig. 7, adding MLP heads improves the performance of other methods, but SSB still greatly outperforms all of them, indicating that the performance improvement of our method can not be merely explained by the increase of the model capacity.

5. Conclusion and Limitations

In this paper, we study a realistic and challenging setting, open-set SSL, where unlabeled data contains outliers from categories that do not appear in the labeled data. We first demonstrate that classifier-confidence-based pseudo-labeling can effectively improve the unlabeled data utilization ratio and leverage useful OOD data, which largely improves the classification performance. We find that adding non-linear transformations between the task-specific head and the shared features provides sufficient decoupling of the two heads, which prevents mutual interference and improves performance in both tasks. Additionally, we propose pseudo-negative mining to improve OOD detection. It uses pseudo-outliers to enhance the representation learning of OOD data, which further improves the model's ability to distinguish between inliers and OOD samples. Overall, we achieve state-of-the-art performance on several benchmark datasets, demonstrating the effectiveness of the proposed method.

Nonetheless, SSB has potential limitations. Despite the improved overall performance, the outlier detector suffers from overfitting as the performance gap between detecting seen outliers and unseen outliers is still very large. Therefore, in the future, more regularizations need to be considered to improve the generalization. Another drawback is that our method is not able to deal with long-tail distributions, which is also very realistic in practice. Presumably, our method will have difficulty distinguishing inliers of tail classes and OOD data due to the data scarcity at tail.

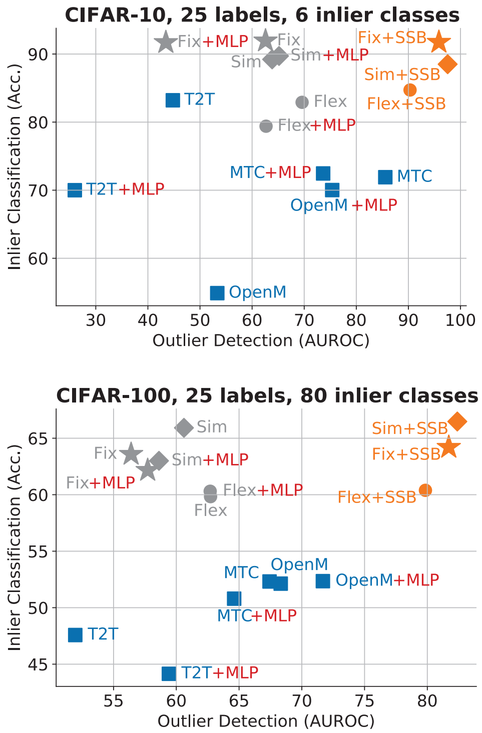


Figure 7: **Comparison between SSB and other methods with the same model parameters.** The performance improvement of SSB can not be simply explained by the increased number of parameters.

References

[1] Charu C Aggarwal and Philip S Yu. Outlier detection for high dimensional data. In *Proceedings of the 2001 ACM SIGMOD international conference on Management of data*, 2001. 1

[2] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020. 2

[3] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *8th International Conference on Learning Representations, ICLR*, 2020. 1, 2

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, 2019. 1, 2

[5] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. In *International Conference on Learning Representations*, 2022. 2

[6] Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning. *arXiv preprint arXiv:2102.03526*, 2021. 2

[7] Yanbei Chen, Xiatian Zhu, Wei Li, and Shaogang Gong. Semi-supervised learning under class distribution mismatch. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 1, 2, 3

[8] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 5

[9] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE conference on computer vision and pattern recognition Workshops*, 2020. 5

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, 2009. 5

[11] Yue Fan, Anna Kukleva, and Bernt Schiele. Revisiting consistency regularization for semi-supervised learning. In *DAGM German Conference on Pattern Recognition*, 2021. 1

[12] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, 2005. 5

[13] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007. 5

[14] Lan-Zhe Guo, Zhen-Yu Zhang, Yuan Jiang, Yu-Feng Li, and Zhi-Hua Zhou. Safe deep semi-supervised learning for unseen-class unlabeled data. In *International Conference on Machine Learning*, 2020. 1, 2, 3

[15] 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*, 2016. 5

[16] Rundong He, Zhongyi Han, Xiankai Lu, and Yilong Yin. Safe-student for safe deep semi-supervised learning with unseen-class unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 3

[17] Rundong He, Zhongyi Han, Yang Yang, and Yilong Yin. Not all parameters should be treated equally: Deep safe semi-supervised learning under class distribution mismatch. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022. 1, 2, 3

[18] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2016. 5

[19] Victoria Hodge and Jim Austin. A survey of outlier detection methodologies. *Artificial intelligence review*, 2004. 1

[20] Junkai Huang, Chaowei Fang, Weikai Chen, Zhenhua Chai, Xiaolin Wei, Pengxu Wei, Liang Lin, and Guanbin Li. Trash to treasure: Harvesting ood data with cross-modal matching for open-set semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 1, 2, 3, 4, 8, 9

[21] Zhuo Huang, Jian Yang, and Chen Gong. They are not completely useless: Towards recycling transferable unlabeled data for class-mismatched semi-supervised learning. *IEEE Transactions on Multimedia*, 2022. 1, 2, 3

[22] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 5

[23] Jack Kiefer and Jacob Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 1952. 5

[24] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images, 2009. Technical report. 5

[25] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR*, 2017. 1, 2

[26] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, 2013. 1, 2

[27] Junnan Li, Caiming Xiong, and Steven C.H. Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[28] Oded Maimon and Lior Rokach. Data mining and knowledge discovery handbook. 2005. 1

[29] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 2018. 1, 2

[30] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010. 5

[31] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *In NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011. 5

[32] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 2008. 5

[33] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in neural information processing systems*, 2018. 1, 2

[34] Jongjin Park, Sukmin Yun, Jongheon Jeong, and Jinwoo Shin. Opencos: Contrastive semi-supervised learning for handling open-set unlabeled data. In *European conference on computer vision Workshop*, 2022. 2

[35] Hieu Pham, Qizhe Xie, Zihang Dai, and Quoc V Le. Meta pseudo labels. *arXiv preprint arXiv:2003.10580*, 2020. 2

[36] Mamshad Nayeem Rizve, Navid Kardan, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Openldn: Learning to discover novel classes for open-world semi-supervised learning. In *European Conference on Computer Vision*, 2022. 2

[37] Mamshad Nayeem Rizve, Navid Kardan, and Mubarak Shah. Towards realistic semi-supervised learning. In *European Conference on Computer Vision*, 2022. 2

[38] Kuniaki Saito, Donghyun Kim, and Kate Saenko. Openmatch: Open-set semi-supervised learning with open-set consistency regularization. *Advances in Neural Information Processing Systems*, 2021. 1, 2, 3, 4, 5, 7, 8, 9

[39] Kuniaki Saito and Kate Saenko. Ovanet: One-vs-all network for universal domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

[40] Robert E Schapire. The strength of weak learnability. *Machine learning*, 1990. 2

[41] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems*, 2020. 1, 2, 3, 5

[42] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, 2017. 1, 2

[43] Hongzhi Wang, Mohamed Jaward Bah, and Mohamed Hammad. Progress in outlier detection techniques: A survey. *Ieee Access*, 2019. 1

[44] Yidong Wang, Hao Chen, Yue Fan, Wang Sun, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, Heli Qi, Zhen Wu, Yu-Feng Li, Satoshi Nakamura, Wei Ye, Marios Savvides, Bhiksha Raj, Takahiro Shinozaki, Bernt Schiele, Jindong Wang, Xing Xie, and Yue Zhang. Usb: A unified semi-supervised learning benchmark for classification. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 1

[45] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019. 1, 2

[46] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2

[47] 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. 5

[48] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *European Conference on Computer Vision*, 2020. 1, 2, 3, 4, 8, 9

[49] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2016. 5

[50] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 2021. 1, 2

[51] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR*, 2018. 2