International Journal of Computer Vision (2024) 132:5888–5904
https://doi.org/10.1007/s11263-024-02155-y

Exploration and Exploitation of Unlabeled Data for Open-Set Semi-supervised Learning

Ganlong Zhao[1] · Guanbin Li[1,2] · Yipeng Qin[3] · Jinjin Zhang[4] · Zhenhua Chai[4] · Xiaolin Wei[4] · Liang Lin[1] · Yizhou Yu[5]

Received: 8 October 2023 / Accepted: 17 June 2024 / Published online: 8 July 2024

Abstract
In this paper, we address a complex but practical scenario in semi-supervised learning (SSL) named open-set SSL, where unlabeled data contain both in-distribution (ID) and out-of-distribution (OOD) samples. Unlike previous methods that only consider ID samples to be useful and aim to filter out OOD ones completely during training, we argue that the exploration and exploitation of ***both*** ID and OOD samples can benefit SSL. To support our claim, (i) we propose a prototype-based clustering and identification algorithm that ***explores*** the inherent similarity and difference among samples at feature level and effectively cluster them around several predefined ID and OOD prototypes, thereby enhancing feature learning and facilitating ID/OOD identification; (ii) we propose an importance-based sampling method that ***exploits*** the difference in importance of each ID and OOD sample to SSL, thereby reducing the sampling bias and improving the training. Our proposed method achieves state-of-the-art in several challenging benchmarks, and improves upon existing SSL methods even when ID samples are totally absent in unlabeled data.

Keywords Semi-supervised learning · Open-set · Image classification

1 Introduction

Semi-supervised learning (SSL) is a promising machine learning approach that exploits unlabeled data to mitigate the costly data labeling process. Given a small set of labeled data and a large set of unlabeled data, SSL aims to train a classifier that surpasses its supervised variant trained only on the labeled dataset. Classic SSL techniques include consistency regularization (Sajjadi et al., 2016; Laine & Aila, 2017), pseudo labeling (*a.k.a.* self-training) (Lee et al., 2013; Pham et al., 2021) and entropy minimization (Grandvalet & Bengio, 2004). Recently, FixMatch (Sohn et al., 2020) achieved state-of-the-art performance by simply combining consistency regularization with pseudo labeling. Although being effective, traditional SSL methods implicitly assumed that the unlabeled data share the same label space with the

Communicated by ZHUN ZHONG.

✉ Guanbin Li
liguanbin@mail.sysu.edu.cn

Ganlong Zhao
zhaogl@connect.hku.hk

Yipeng Qin
QinY16@cardiff.ac.uk

Jinjin Zhang
zhangjinjin05@meituan.com

Zhenhua Chai
chaizhenhua@meituan.com

Xiaolin Wei
weixiaolin02@meituan.com

Liang Lin
linliang@ieee.org

Yizhou Yu
yizhouy@acm.org

[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China

[2] Research Institute, Sun Yat-sen University, Shenzhen, China

[3] School of Computer Science and Informatics, Cardiff University, Cardiff, UK

[4] MeituanDianping Group, Beijing, China

[5] Department of Computer Science, The University of Hong Kong, Pok Fu Lam, Hong Kong

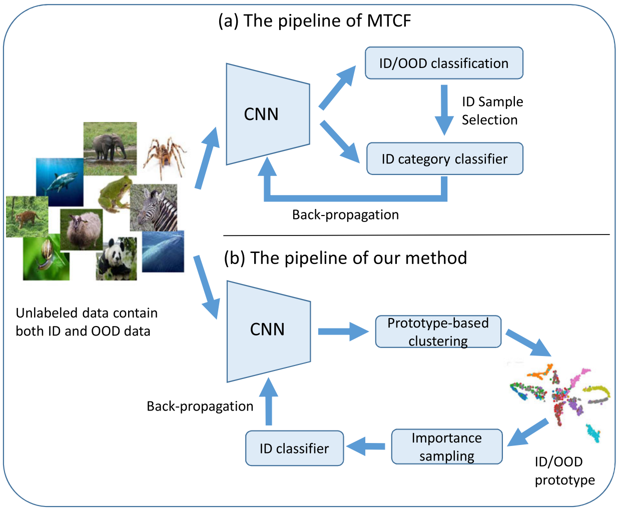


Fig. 1 The comparison between our method and MTCF (Yu et al., 2020). MTCF sets an independent OOD detection branch to the backbone, and performs ID/OOD classification on the features. OOD samples are excluded from semi-supervised learning. Different from MTCF, our method first clusters both ID and OOD features to prototypes. Prototypes are assigned ID/OOD labels and unlabeled samples are sampled by their importance and prototypes. Our method exploits both ID and OOD data for the backbone training

labeled data during training, which limits their application in the open-set real-world scenarios.

Open-set SSL extends SSL to open-set datasets where the unlabeled data contain both in-distribution (ID) and out-of-distribution (OOD) samples. Specifically, ID samples share the same label space with labeled data while OOD samples may be out of that label space. Yu et al. (2020) pioneered this direction and proposed to eliminate the negative effects of OOD samples using an OOD detector (Liang et al., 2018). Thanks to the OOD detector, they identified high-confidence ID samples and gradually incorporated them into the training of a MixMatch (Berthelot et al., 2019) model with their multi-task curriculum framework (Fig. 1).

Although MTCF (Yu et al., 2020) is effective, we argue that it has the following two shortcomings. First, it overlooks the role of OOD samples in feature learning. In their method, OOD samples are excluded from SSL whereas we argue that *if being properly used, OOD samples can benefit feature learning and thus SSL*, especially when there are few ID samples in the unlabelled dataset. Second, their method depends on the performance of its OOD detector and thus performs poorly on high-variance datasets where the ambiguity between ID and OOD samples makes it prone to misclassification. As pointed out by previous studies (Winkens et al., 2020), near-OOD tasks where OOD samples are close to ID ones can greatly lower the performance of OOD detection method. Simply filtering out all OOD samples can be difficult and thus degrades the performance of semi-supervised training when OOD samples dominate the unlabeled dataset. In addition, their evaluation is based on synthetic OOD samples (Yu et al., 2020) (e.g. Gaussian noise, Uniform noise) and images of completely irrelevant topics, which may not generalize to real-world scenarios where OOD samples can be "close" to ID ones.

In previous semi-supervised studies, pseudo labeling is an important technique that can utilize the unlabeled data and thus improve the performance of semi-supervised methods. Pseudo labeling encourages the model to output high-confidence prediction for unlabeled samples and thus construct a better feature extractor (Sohn et al., 2020). However, if unlabeled data contains both ID and OOD images, pseudo-labeling-based methods will force ID and OOD samples with the same label prediction to get closer, which degrades the performance of the feature extractor and the accuracy of ID/OOD classification. Therefore, our method aims to construct and preserve the inner structure of both ID and OOD features to train a better feature extractor and to facilitate the ID/OOD classification.

In this paper, we address the aforementioned shortcomings of open-set SSL by exploring and exploiting the unlabeled data including both ID and OOD samples (Fig. 2). Specifically, we first propose a prototype-based clustering and identification algorithm that clarifies the ambiguity between ID and OOD samples by *exploring* the inherent similarity and difference among their features, and thus better identifies the unlabeled samples. Then, we propose a novel importance sampling method that reduces the sampling bias by *exploiting* the difference in importance of each ID and OOD sample to SSL, thereby improving training. We implement this method with our newly proposed cascading pooling strategy, which increases the density of ID samples in mini-batches and further stabilizes training. Empirically, we verify the effectiveness of our method on three standard benchmark datasets (CIFAR-100 (Krizhevsky et al., 2009), SVHN (Netzer et al., 2011) and TinyImageNet (Deng et al., 2009)) and a new dataset, DomainNet-Real (Peng et al., 2019), which is more challenging and realistic. In summary, our contributions include:

- We demonstrate that the performance of open-set semi-supervised learning (SSL) can be improved by utilizing out-of-distribution (OOD) samples.
- We design a novel prototype-based clustering and identification algorithm and demonstrate its effectiveness in feature learning.
- We propose a new importance-based sampling method that reduces sampling bias and improves training.
- We verify the effectiveness of our method on larger and more challenging benchmarks including DomainNet-Real and ImageNet (Deng et al., 2009). Extensive experimental results on these benchmark datasets demonstrate the superiority of our proposed method.

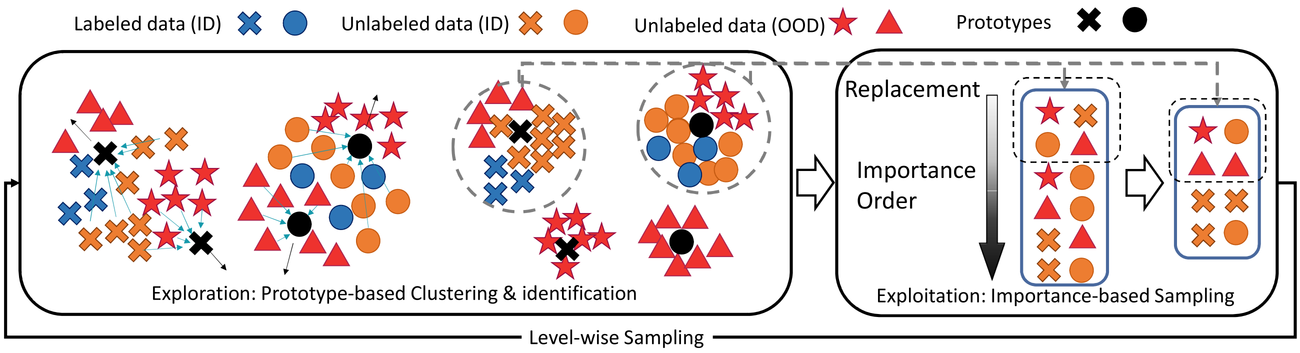


Fig. 2 Exploration and exploitation of the unlabeled data including in-distribution (ID) and out-of-distribution (OOD) samples. Exploration: we cluster ID and OOD samples in each mini-batch and identify them accordingly based on an exploration of the inherent similarity and difference of their features (represented by prototypes). Exploitation: we exploit the different importance of each ID and OOD sample in SSL to reduce the sampling bias during training

2 Related Works

Semi-Supervised Learning (SSL) addresses the scarcity of labeled data by leveraging the relationship between a small amount of labeled data and a large amount of unlabeled data. In general, two common SSL techniques that are widely applied to semi-supervised learning are consistency regularization and pseudo labeling (*a.k.a.* self-training). Consistency regularization (CR) (Laine & Aila, 2017; Bachman et al., 2014; Sajjadi et al., 2016; Zhai et al., 2019) assumes that the classification results should only rely on the semantics of input images, and penalizes the change of model outputs against the perturbation or augmentation of input images. Some CR methods employ adversarial perturbation or dropout (Park et al., 2018; Wager et al., 2013) on the input images while data augmentation (Berthelot et al., 2019; Sajjadi et al., 2016) is widely recognized to be more effective. From another perspective, pseudo labeling (Lee et al., 2013; Pham et al., 2021) assigns pseudo labels to unlabeled data according to the model's prediction confidence and steers its own training with those pseudo labels. FixMatch (Sohn et al., 2020) combines the ideas of pseudo labeling and consistency regularization, and achieves state-of-the-art performance on several benchmarks for semi-supervised learning. FixMatch utilizes two different augmentations of the input image, strong augmentation, and weak augmentation, and trains the model with the strong-augmented images and the pseudo labels generated by corresponding weak-augmented images. Similar to pseudo labeling, Entropy minimization (Grandvalet & Bengio, 2004) encourages the model to output low-entropy (i.e. high confidence) prediction for unlabeled samples. Besides, there have been other techniques for semi-supervised learning. Temporal ensemble (Laine & Aila, 2017) forms a consensus prediction for the unlabeled data using the outputs of the network-in-training on different epochs. Mean teacher (Tarvainen & Valpola, 2017) averages model weights instead of label predictions to avoid the problem that temporal ensemble becomes unwieldy when learning from large datasets. FlexMatch (Zhang et al., 2021) proposes a curriculum learning approach for semi-supervised learning to leverage unlabeled data according to the model's learning status. Some self-supervised methods (Li et al., 2020) also employ prototype-based methods for semi-supervised learning, however, their clustering strategies are purely unsupervised and not applicable to OOD detection during training.

Out-Of-Distribution (OOD) Identification (Liang et al., 2018; DeVries & Taylor, 2018; Ming et al., 2022; Du et al., 2022; Yang et al., xxx) aims to identify the OOD samples in a given dataset which consists of both In-Distribution classes and Out-Of-Distribution samples. For image classification, conventional methods like density estimation or nearest neighbor (Chow, 1970; Vincent & Bengio, 2003; Ghoting et al., 2008) are not applicable due to the high dimensionality of image feature space. Addressing this issue, DNN-based OOD detectors (Liang et al., 2018; Hendrycks & Gimpel, 2017) have been proposed. Based on the observation that ID samples tend to have higher softmax scores, Hendrycks and Gimpel (2017) propose a baseline method for OOD detection without retraining networks. Liang et al. (2018) improve such a baseline by introducing temperature scaling in the softmax function to increase the softmax score gap between ID and OOD samples. The difficulty of the OOD detection depends on how semantically close to the inlier classes, i.e., ID classes are to the outliers, i.e., OOD samples. Winkens et al. (2020) distinguish the difficulty difference between near-OOD tasks and far-OOD tasks by the difference of state-of-the-art performance for area under the receiver operating characteristic curve (AUROC). Some methods (Lee et al., 2018; Liu et al., 2020; Hsu et al., 2020; Sun et al., 2021) tackle the OOD detection problem by class conditional Gaussian distributions, energy function or rectified activations. However, most of them detect the OOD samples post hoc, which is not suitable for open-set semi-supervised learning.

Open-Set Semi-Supervised Learning (Oliver et al., 2018; Yu et al., 2020; Luo et al., 2021; Chen et al., 2020; Huang et al., 2021; Park et al., 2022; He et al., 2022a, b; Fan et al., 2023; Huang et al., 2022) aims to develop robust SSL algorithms which work on "dirty" unlabeled data that contain OOD samples. Oliver et al. (2018) first pointed out that the performance of SSL techniques can degrade drastically when the unlabeled data contain a different distribution of classes. This inspires MTCF (Yu et al., 2020) which incorporates an OOD detection branch to MixMatch (Berthelot et al., 2019) and works by gradually adding high-confidence ID samples to semi-supervised training. However, it ignores the contribution of consistency regularization to SSL, which is independent to OOD detection. From this perspective, OOD samples are harmless and can even be beneficial. Thus, excluding them from training may not be the optimal solution and can be impractical for big datasets containing a large proportion of OOD samples. To this end, we propose to utilize OOD samples instead of filtering them out during training. As a concurrent work, (Luo et al., 2021) viewed the categorical difference between OOD and ID samples as a distributional difference and attempted to reduce the distribution divergence using style transfer. They also explored the OOD samples during training via unsupervised data augmentation (Xie et al., 2020). UASD (Chen et al., 2020) tackled a problem called *Class Distribution MisMatch* where some classes in the labeled data are absent in the unlabeled data, and vice versa. Although looks similar, this problem is different from ours. Huang et al. (2021) propose a cross-modal matching strategy to detect OOD samples and train the network to match samples to an assigned one-hot class label. Besides, some other works study the problem of novel category discovery (Vaze et al., 2022; Wen et al., 2023; Zhang et al., 2023; An et al., 2023) from unlabeled data, which is a similar setting to open-set semi-supervised learning but focuses on novel class discovery and estimation with clustering accuracy.

3 Preliminary

Given a small labeled dataset $\mathcal{X} = \{(x_i, y_i)\}_{i=1}^{N_l}$ and a large unlabeled dataset $\hat{\mathcal{X}} = \{\hat{x}_i\}_{i=1}^{N}$ where $N \gg N_l$ and $y_i \in (1, \ldots S)$, semi-supervised learning for classification aims to learn a model that performs best by utilizing both $\mathcal{X}$ and $\hat{\mathcal{X}}$. Different from traditional semi-supervised learning, open-set semi-supervised learning aims to utilize an unlabeled dataset $\hat{\mathcal{X}}$ containing out-of-distribution samples whose ground truth labels are not in $(1, \ldots S)$. Our method aims to train a model to achieve higher accuracy on the test set which contains only in-distribution data.

4 Method

Our method has two components: (i) a prototype-based clustering and identification algorithm that learns better representations for the identification of In-Distribution (ID) and Out-Of-Distribution (OOD) samples by clustering them in an unsupervised way; (ii) an importance sampling method that samples unlabeled data according to their importance to SSL, thereby reducing the sampling bias and improving the training (Fig. 4). Specifically, our clustering and identification algorithm helps pseudo-labeling by pushing ambiguous ID and OOD samples away from each other (towards different prototypes) in the feature space. Note that as an unsupervised representation learning method, our clustering process benefits a lot from the OOD data that "augment" the dataset. The resulting clusters can be binarily identified as ID and OOD ones according to labeled data. Based on the identification, we design a novel importance sampling method that assigns importance scores to unlabeled data and samples them accordingly. This addresses the problem of random sampling where early-identified ID samples are over-sampled while later ones are under-sampled. Furthermore, we devise a cascading pooling strategy to improve the density of ID samples in mini-batch training, which further stabilizes the training. The overview of our method is shown in Algorithm 1.

Algorithm 1 Overview of our method.

```
Require: initialized prototypes
Ensure: clustering loss $L_c$, semi-supervised loss $L_{SSL}$, class number $S$
for number of training iterations do
    Sample a minibatch of labeled samples
    Sample a minibatch of unlabeled samples from our pyramid of sample pools
    Compute semi-supervised learning loss $L_{SSL}$ for labeled and unlabeled minibatches
    for each sample do
        for each class $s \in \{1, 2, \ldots, S\}$ do
            Compute $L_c$ (Eq. 2) and $L_{labeled}^{y}$ (Eq. 3) for unlabeled samples and labeled samples of class $s$ respectively using our prototype-based clustering algorithm
            Update the prototypes using Eq. 5
            Update the sample pools accordingly
        end for
    end for
end for
```

4.1 Prototype-Based Clustering and Identification

As Fig. 3 (top row) shows, ID and OOD samples are mixed in unlabeled data. A key defect of open-set pseudo labeling is that OOD samples can easily be misclassified as ID samples, thereby confusing the feature extractor. Addressing this issue, we propose a prototype-based clustering algorithm to clarify the ambiguity between ID and OOD samples (Fig. 3, bottom row). Let F be an SSL classifier, $\hat{x}_i$ be a sample in the unlabeled dataset $\hat{\mathcal{X}} = \{\hat{x}_i\}_{i=1}^{N}$, $\hat{s}$ be the pseudo label of $\hat{x}_i$ assigned by F, $F(\hat{x}_i)$ be the output probabilities of F with input $\hat{x}_i$, and $f(\hat{x}_i)$ be the normalized feature extracted by f (a subnetwork of F, *a.k.a.* a feature extractor), our clustering algorithm is detailed as follows:

Prototype Initialization. This step aims to set up K initial prototypes p_j^s, $j \in \{1, 2, \ldots, K\}$ for each class $s \in \{1, 2, \ldots, S\}$. First, we pretrain F until each class s contains at least L unlabelled samples, $L \ll N/S$ where N is the unlabeled set size. These samples are assigned pseudo labels $\hat{s}$. Then, for each class s, we extract the features of all its unlabeled samples by f and initialize our prototypes p_j^s (Fig. 3, black marks) as the k-means cluster centers of the extracted features. In this step, both K and L are hyperparameters (Fig. 4).

Clustering Loss. For each unlabeled sample $\hat{x}_i$ in a mini-batch during training, given its pseudo label $\hat{s}$ (generated by SSL method) and the associated K prototypes p_j^s, $j \in \{1, 2, \ldots, K\}$ of class s, we define our prototype-based clustering loss as:

$$L_c^s(\hat{x}_i) = - \log \frac{\exp(f(\hat{x}_i) \cdot p_*^s/\tau)}{\sum_{j=1}^{K} \exp(f(\hat{x}_i) \cdot p_j^s/\tau)} \mathbb{1}(\max(F(\hat{x}_i)) > t_c), \tag{1}$$

where $\mathbb{1}$ is an indicator function, t_c is a threshold parameter, p_*^s is the prototype that is closest to $f(\hat{x}_i)$ in Euclidean space, τ is a temperature parameter used in self-supervised learning (Chen et al., 2020). Following previous studies (Chen et al., 2020; He et al., 2020; Wu et al., 2018), both $f(\hat{x}_i)$ and p_*^s are normalized, and thus $f(\hat{x}_i) \cdot p_*^s$ is the cosine similarity between them. Intuitively, minimizing $L_c^s(\hat{x}_i)$ guides classifier F to generate features $f(\hat{x}_i)$ closer to p_*^s but further to other prototypes. Thus, the overall loss across a mini-batch of unlabeled samples is:

$$L_c = \sum_{j=1}^{BS} L_c^s(\hat{x}_j) \tag{2}$$

where BS is the batch size. L_c is used as an additional loss term in the SSL loss function.

In addition to the clustering loss for unlabeled data, we further apply a similar loss to the labeled data, clustering them to the same centers. Specifically, given a labeled sample x_i $(1 \leq i \leq N_y)$ and its ground truth label y, where N_y is the total number of labeled samples with label y, we define the loss as:

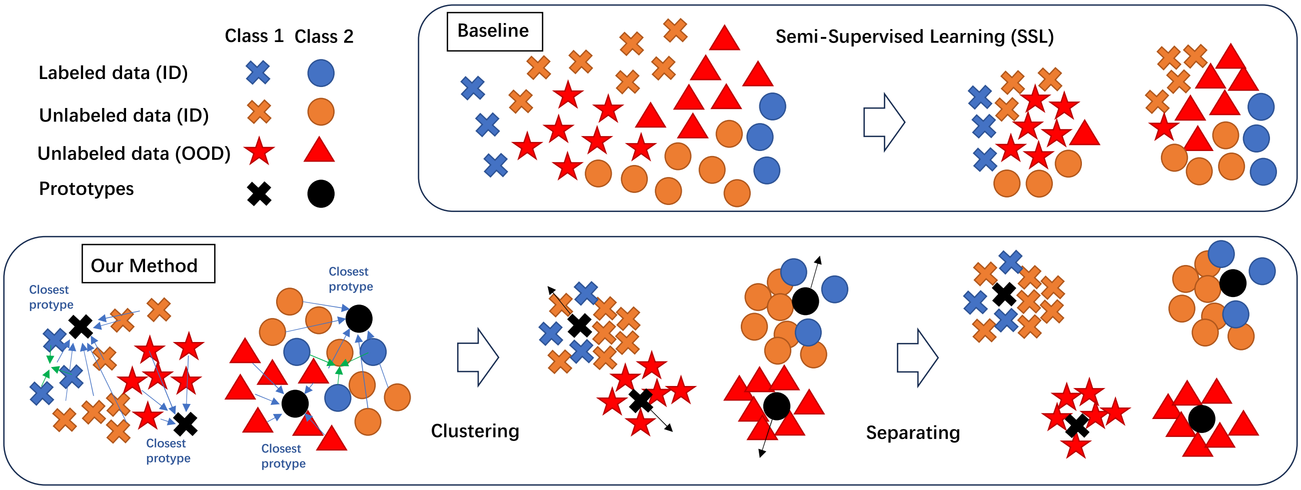


Fig. 3 Illustration of our prototype-based clustering method. Top row: Semi-supervised learning (SSL) methods might be confused by the OOD samples and incorrectly assign in-distribution (ID) pseudo labels to them, which degrades their performance. Bottom row: With our clustering method, ID and OOD samples are pushed away from each other towards a set of pre-defined prototypes (black marks), which clarifies the ambiguity between ID and OOD samples and facilitates ID/OOD identification. The blue and green marks represent the unlabeled data and labeled data clustering respectively. The positions of the prototypes are dynamically updated during training

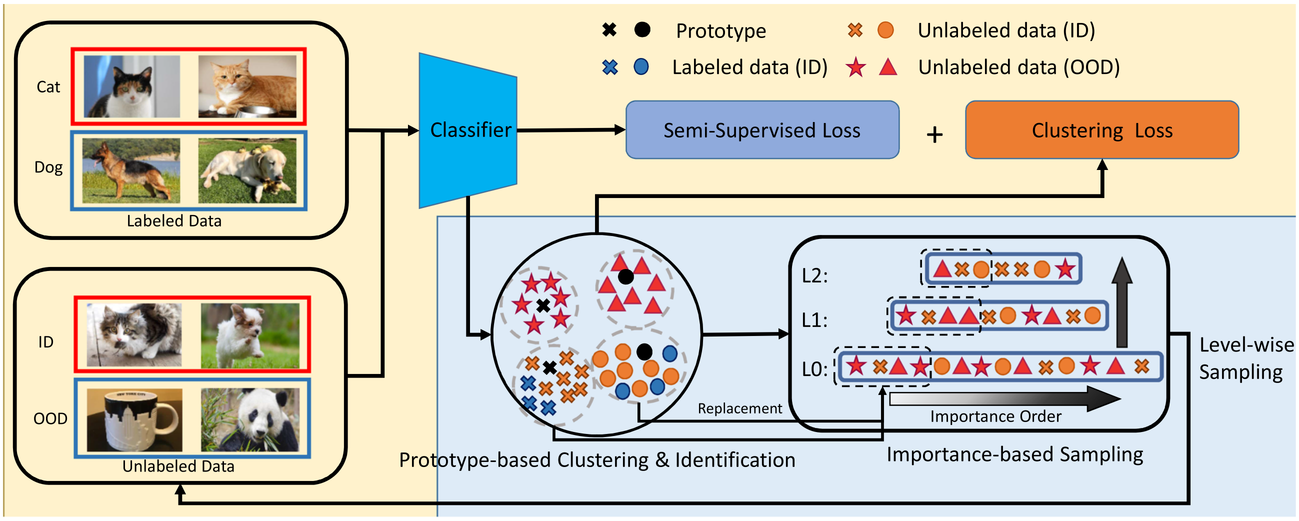


Fig. 4 Overview of our method. We propose two techniques, a prototype-based clustering and identification algorithm, and an importance-based sampling method to improve the performance of open-set semi-supervised learning (SSL). Our clustering and identification algorithm clusters samples at the feature level and thus facilitates feature learning by increasing the distances between ID and OOD samples. Our importance-based sampling method facilitates SSL by reducing the sampling bias during training

$$L^{y}_{labeled}(x_i) = -f(x_i) \cdot q_y + L^{s}_{c}(x_i), \tag{3}$$

where q_y is the normalized feature center of class y:

$$q_y = \text{normalize}(\sum_{i=1}^{N_y} f(x_i)). \tag{4}$$

In Eq. 3, the first term aims to cluster all labeled samples of the same class towards their normalized feature center q_y, preventing them from being misaligned to different ID/OOD centers; the second term applies the same clustering loss for unlabeled data (Eq. 1) to labeled data, indicating that both the labeled and unlabeled ID samples should be clustered to the same centers. The clustering loss clusters the features of both labeled and unlabeled data in a semi-supervised manner, thus separates the ID and OOD data into different cluster centers. *Prototype Update.* At the same time, for each sample $\hat{x}_i$ in a mini-batch during training, we dynamically update its nearest prototype p^s_* as a moving average:

$$p^s_{*,(t+1)} = \text{normalize}(\alpha p^s_{*,(t)} + \beta f(\hat{x}_i)), \tag{5}$$

where $\alpha = 0.99$ and $\beta = 0.01$ are weighting parameters following the common practice of momentum update.

With our prototype-based clustering algorithm, pseudo-labeled samples in each class s are clustered according to the similarity of their features. As a result, heterogeneous samples are pushed away from each other. This helps SSL as the difference between OOD and ID samples are also clarified, thereby helping the feature extractor to learn better representations. Thus, OOD samples are less likely to be misclassified as ID samples and damage the self-training. Based on the clustering results, we identify ID/OOD samples as follows. *Sample Identification.* First, we identify the *unlabeled* ID samples to be included in the pools according to their distances to the *labeled* ID samples in the labeled dataset $\mathcal{X} = \{(x_i, s_i)\}_{i=1}^{N_l}, s_i \in \{1, 2, \dots, S\}$ where S is the number of classes. Let $f(x_i)$ be the normalized feature of x_i that is extracted by f, we can calculate the per-class feature centers of the labeled data as:

$$O_s = \frac{\sum_{i=1}^{N_l} \mathbb{1}(s_i = s) f(x_i)}{\sum_{i=1}^{N_l} \mathbb{1}(s_i = s)}, s \in \{1, 2, \dots, S\} \tag{6}$$

where $\mathbb{1}$ is an indicator function. Since all labeled images are ID samples, an unlabeled sample with pseudo label s tends to be ID if its corresponding prototype p^s_* is close to O_s. Therefore, for each class s, we compute the Euclidean distances from O_s to each of its prototypes p^s_j. According to these distances, we sort all prototypes p^s_j in increasing order and pick the first N_{id} of them as ID prototypes. For each unlabeled sample $\hat{x}_i$ in a mini-batch, we identify it as ID if its closest prototype p^s_* is an ID prototype. Otherwise, $\hat{x}_i$ is identified as OOD. In this step, N_{id} is a hyperparameter.

4.2 Importance Sampling for Open-Set SSL

Recalling the definition of Open-set SSL where the dataset contains two types of samples, i.e. ID and OOD samples, it is straightforward to assume that they are of different importance to SSL: ID samples are useful while OOD ones are less relevant for the target task. Such an assumption motivates the selection of ID samples in SSL that is widely employed in previous methods. Despite that OOD samples can help the training of semi-supervised methods, overwhelming OOD samples in mini-batches will unstabilize the training and lower the performance when the number of images are large and the batch size is small. Therefore, We follow the ID sample selection paradigm and try to improve the performance by making the classifier concentrate on ID samples while utilizing the information of OOD samples. Unlike previous methods, we noticed that ID samples are dynamically identified during training. Thus, the ID samples identified earlier occur more often in random sampling and are thus biased. This is undesirable as they can soon be well-learned and contribute less to the training than newly identified ones. To this end, we propose a novel importance sampling method for mini-batch sampling during training that assigns importance scores to unlabeled samples and only maintains the important ones in the sample pools as follows.

Importance-based Sample Pools. After identification, the identified OOD samples are assigned importance scores of 0 and excluded from SSL; the identified ID samples are assigned importance scores of $1/I(\hat{x}_i)$ where $I(\hat{x}_i)$ is the number of times $\hat{x}_i$ is identified as an ID sample throughout the training. We maintain the identified ID samples in our per-class importance-based sample pools P_s, where $s \in \{1, 2, \ldots, S\}$ is the class label. We restrict P_s to contain at most N_p samples, $N_p \gg BS$ where BS is the batch size. During mini-batch training, assume that N_s ID samples are identified for class s in one iteration, we update P_s by:

- *Case 1.* If P_s has enough space, we simply add the N_s ID samples to P_s.

- *Case 2.* Otherwise, assuming that the shortfall of empty space for N_s ID samples is M, we first add the $N_s - M$ ID samples to P_s. For the rest M ID samples, we compute probability $\mathcal{P}_{\hat{x}_i}$ for each sample $\hat{x}_i$ in P_s, $i \in \{1, 2, \ldots, N_P\}$ using their importance scores as:

$$\mathcal{P}_{\hat{x}_i} = \min\{M \frac{I(\hat{x}_i)}{\sum_{j=1}^{N_p} I(\hat{x}_j)}, 1\}. \tag{7}$$

Then, we select each sample by probability $\mathcal{P}_{\hat{x}_i}$ and obtain N_r samples. Please note that we multiply the probability by M because we try to select M samples from the sample pool P_s. We replace the first $\min(N_r, M)$ of them with the newly identified ID samples.

Intuitively, an ID sample is more likely to be removed from P_s if it is sampled more often, i.e., it is well-learned.

However, it is difficult to identify ID samples accurately by performing the identification once when the unlabeled set is complicated and OOD data can benefit the semi-supervised training. Besides, the density of ID samples in a mini-batch is important for the performance as we show in Sect. 5.4. To this end, we devise a cascading pooling strategy to further improve the density of ID samples as follows, and it can help to stabilize the SSL training by providing high-density ID samples within a mini-batch.

Cascading Sample Pools. Let S be the number of classes, we cascade different sets of sample pools as a pyramid:

- Level 0 of the pyramid is the raw dataset.
- Level 1 is a set of S ID sample pools. The capacity of each sample pool is N_P.
- Level 2 is a set of S ID sample pools. The capacity of each sample pool is $N_P/2$.
- ......
- Level N is a set of S ID sample pools. The capacity of each sample pool is $N_P/2^{N-1}$.

During training, we circularly draw mini-batches of samples in a level-wise manner from Level 0 to Level N. In each training iteration, we draw samples evenly from the S sample pools in the same level and apply ID sample identification to it. The newly identified ID samples are used to update the sample pools at the next level.

5 Experiment

5.1 Experimental Setup

Datasets. Following the common practice in SSL evaluation (Sohn et al., 2020), we test our method on four benchmark datasets:

- *CIFAR-100* (Krizhevsky et al., 2009): a dataset consisting of 100 classes of natural images. Each class contains 500 training images and 100 testing images.

- *SVHN* (Netzer et al., 2011): a dataset consisting of 10 classes of digits images. It contains 73,257 and 26,032 digits images for training and testing respectively.

- *TinyImageNet*: a subset of the ImageNet dataset (Deng et al., 2009) consisting of 200 classes of natural images. Each class contains 500 images for training and 50 images for testing.

And a more challenging and realistic dataset:

- *DomainNet-Real* (Peng et al., 2019): DomainNet is a dataset consisting of 345 classes of images in 6 domains (e.g. real, painting, sketch). In our experiments, we only use the 172,947 images in its Real domain as we observed that FixMatch (Sohn et al., 2020) performs poorly in some domains.

Implementation Details. We implement our method on top of FixMatch (Sohn et al., 2020), a state-of-the-art SSL algorithm. In addition to the relatively standard pseudo labeling, FixMatch used another common SSL technique: consistency regularization. In a nutshell, it encourages the SSL classifier to output the same value for two variants of an unlabeled sample $\hat{x}_i$: a weak-augmented variant $\hat{x}_i^a$ and a strong-augmented variant $\hat{x}_i^b$. Accordingly, for class s and sample $\hat{x}_i$, we extend L_c^s (Eq. 1) to $L_c^{s,\text{CR}}$ as follows:

$$L_c^{s,\text{CR}}(\hat{x}_i) = L_c^s(\hat{x}_i^a) + L_c^s(\hat{x}_i^b) \tag{8}$$

Note that we use the same target prototype p_*^s that is closest to the weak-augmented variant $\hat{x}_i^a$ for both $L_c^s(\hat{x}_i^a)$ and $L_c^s(\hat{x}_i^b)$ because (i) heuristically, $\hat{x}_i^a$ is weak-augmented and thus closer to $\hat{x}_i$ in the feature space; (ii) in line with consistency regularization, $\hat{x}_i^a$ and $\hat{x}_i^b$ share the same semantic meanings and should be in the same cluster, i.e. with the same prototype. Similarly, we only use the weak-augmented variants of unlabeled samples in prototype update and sample identification. Following FixMatch, we employ different network architectures for different datasets. We tune the hyper-parameters using a small validation set.

- For CIFAR-100, SVHN and TinyImageNet, we follow FixMatch (Sohn et al., 2020) and use the same architecture based on Wide ResNet (WRN 28×8) (Zagoruyko & Komodakis, 2016). All images are resized to 32 × 32. We set the number of prototypes $K = 10$ and the weight of L_c as 0.01 when added to the FixMatch loss function. Following (Chen et al., 2020), we set $\tau = 0.07$ and $t_c = 0.98$, which is slightly higher than FixMatch's pseudo labeling threshold of 0.95. We use the same hyperparameters of FixMatch (Sohn et al., 2020) in the semi-supervised learning (SSL) part of our method. We run our method on 1 Nvidia Tesla V100 GPU with 16GB memory and set the batch size as 64 for labeled data and 448 (64×7) for unlabeled data. We report the experimental results after 100 epochs of training.
- For DomainNet-Real, we use the ResNet-50 (He et al., 2016) architecture. All images are resized to 224×224. We set the number of prototypes $K = 30$ and the weight of L_c as 0.1 when added to the FixMatch loss function. Following the ImageNet (Deng et al., 2009) training scheme in FixMatch (Sohn et al., 2020), we set $t_c = 0.7$ which equals to FixMatch's pseudo labeling threshold. In our experiment, $N_{id} = K/5$. We use the same hyperparameters of FixMatch (Sohn et al., 2020). We run our method on 6 Nvidia Tesla V100 GPUs and report the experimental results after 100 epochs of training. For each GPU, we set the batch size as 8 for labeled data and 56 for unlabeled data. Following FixMatch (Sohn et al., 2020), we apply linear warmup to the learning rate for the first 5 epochs of training until it reaches an initial value of 0.4.
  At epoch 60, we decay the learning rate by multiplying it by 0.1.

Experimental Settings. (1) ID *vs.* OOD. For CIFAR-100, TinyImageNet and SVHN, *ID samples* are defined as the images in the first N classes; *OOD samples* are defined as those in the rest classes. For DomainNet-Real, *ID samples* are defined as the images in the N classes with the most images; *OOD samples* are defined as the 50k images sampled from the rest classes, which aims to balance the numbers of ID and OOD samples. (2) Labeled *vs.* unlabeled. For all datasets, *labeled data* is defined as the first 25 images and their associated labels in each of the N classes; *unlabeled data* is defined as the rest images in each of the N classes together with the OOD samples. (3) Training *vs.* Testing. For DomainNet, for each of the N classes, the *testing set* is defined as the 100 images sampled from the unlabeled data in the class; For the other three datasets, we directly use the predefined testing set; the *training set* is defined as other images (including both labeled and unlabeled data) in the class. Furthermore, we report the average performance of the last 10 epochs over 3 runs using the same set of random seeds. The 3 runs use different random seeds.

5.2 Experimental Result

As Table 1 shows, our method significantly outperforms previous open-set semi-supervised learning and OOD detection methods including MTCF (Yu et al., 2020), DS^{3}L (Guo et al., 2020), Energy (Liu et al., 2020), ReAct (Sun et al., 2021) and OpenMatch (Saito et al., 2021). To give a better idea of how well our method performs, we provide two additional baselines using FixMatch (Sohn et al., 2020):

- "Labeled Only": a FixMatch model trained with labeled data only, which can be viewed as a lower bound.
- "Clean": a FixMatch model trained with ID samples only, which can be viewed as an improved baseline.

We test all methods on three datasets: CIFAR-100, TinyImageNet and DomainNet-Real.[1] As discussed in the "Experimental Setup" section, for each dataset, we define the images in its first N ($N = 10, 20$) classes as the ID samples; the OOD samples are defined accordingly.

- For CIFAR-100 and TinyImageNet, we observed small gaps between FixMatch and Clean, which leaves small room for improvement. Similar to Yu et al. (2020), we conjecture that the reason is the relatively simple datasets being used. However, it is interesting to see that our method outperforms "Clean" on CIFAR-100, 10/90 and 20/80 (10/20 ID classes and 90/80 OOD classes from CIFAR-100). This implies that OOD samples are also

[1] We did not use SVHN because it has only 10 classes and thus cannot fit into this experiment.

Table 1 Experimental results on Open-Set Semi-Supervised Learning

Datasets	DomainNet-Real		CIFAR-100		TinyImageNet	
ID/OOD	10/50k*	20/50k*	10/90	20/80	10/190	20/180
Labeled Only	48.5 ± 1.0	41.6 ± 0.7	47.3 ± 1.8	40.0 ± 0.4	36.9 ± 2.3	32.2 ± 0.9
FixMatch (Sohn et al., 2020)	52.8 ± 2.9	49.7 ± 2.5	80.8 ± 0.9	72.2 ± 0.2	68.9 ± 0.7	53.6 ± 1.0
MTCF (Yu et al., 2020)	54.2 ± 1.8	46.3 ± 0.4	59.8 ± 0.6	46.2 ± 1.0	52.4 ± 1.2	46.5 ± 0.6
DS^{3}L (Guo et al., 2020)†	–	–	57.0 ± 0.7	40.2 ± 1.0	52.2 ± 2.7	40.0 ± 1.6
Energy (Liu et al., 2020)	50.1 ± 1.8	45.9 ± 1.0	82.5 ± 0.7	72.9 ± 1.6	67.3 ± 2.0	56.5 ± 1.5
ReAct (Sun et al., 2021)	50.1 ± 1.1	46.6 ± 0.7	82.9 ± 0.7	73.3 ± 2.0	69.5 ± 1.7	57.7 ± 2.0
OpenMatch (Saito et al., 2021)	54.8 ± 2.6	50.4 ± 1.2	83.0 ± 1.0	73.3 ± 2.5	68.7 ± 2.8	54.8 ± 1.0
Ours	**59.4 ± 0.3**	**54.3 ± 1.2**	**85.5 ± 0.8**	**76.0 ± 1.1**	**71.4 ± 0.7**	**58.5 ± 1.1**
Clean	63.5 ± 0.7	60.7 ± 0.8	84.8 ± 0.7	72.3 ± 0.4	79.5 ± 0.8	60.3 ± 0.3

Our method outperforms MTCF (Yu et al., 2020) and improves FixMatch (Sohn et al., 2020) by a significant margin. ID/OOD: the number of classes whose images are defined as ID and OOD samples respectively. $(\cdot)$/50k*: to balance the numbers of ID and OOD samples, we sample 50k images from the classes other than $(\cdot)$ in DomainNet-Real as OOD samples. †: DS^{3}L consumes too much memory and time on DomainNet-Real and cannot run on commodity workstations

Table 2 Experiment results on ImageNet with 50/950 and 100/900 ID/OOD class settings

Dataset	ImageNet	
ID/OOD	50/950	100/900
FixMatch (Sohn et al., 2020)	29.6 ± 0.6	28.1 ± 0.9
Mask-OOD	31.6 ± 0.8	29.4 ± 0.6
SimCLR-OOD	32.8 ± 0.4	30.7 ± 0.9
Ours	33.3 ± 0.6	30.4 ± 1.0
Clean	40.1 ± 0.9	39.7 ± 0.5

useful in SSL, which contradicts the common belief that OOD samples are harmful.

- For DomainNet-Real, we observed approximately 10% gaps between FixMatch and Clean. In such challenging scenarios, our method also significantly outperforms MTCF (Yu et al., 2020) and FixMatch (Sohn et al., 2020). However, there is a considerable gap between our method and "Clean", which suggests that there is still room for improvement.

In summary, experimental results show that our method performs the best against competing methods in all six settings (two for each dataset), which indicates that the improvement brought by our method can be generalized to a variety of datasets and ID/OOD ratios.

5.3 Experimental Result on ImageNet

To investigate the effectiveness of our method on larger and more complicated datasets, we further applied our method to ImageNet (Deng et al., 2009) under the open-set semi-supervised setting. The experiments on ImageNet are conducted in two different settings, 50 ID classes, and 100 ID classes respectively. More specifically, we first randomly select 50/100 classes from all 1000 categories, then use the rest as OOD classes. To align with the experiments in Table 1, each ID class has 25 labeled samples. We adopt the same hyper-parameter settings as the experiments on DomainNet-Real, which are consistent with the setting of FixMatch on ImageNet. For the 50/950 ID/OOD class experiment, the FixMatch baseline achieves 29.6% ± 0.6 and the Clean model achieves 40.1% ± 0.9. Our method improves the performance by 3.7% and gets 33.3% ± 0.6 accuracy. For the 100/900 ID/OOD class experiment, the FixMatch baseline achieves 28.1% ± 0.9 and the Clean model achieves 39.7% ± 0.5. Our method improves the performance by 2.3% and achieves 30.4% ± 1.0. The experimental results demonstrate that our method can be applied to larger datasets with millions of OOD samples.

We further provide the performance of two variants of FixMatch, Masked-OOD and SimCLR-OOD in Table 2. The two variant models are used to verify the motivation of our method and study the effectiveness of OOD sample utilization. More specifically, The Masked-OOD model set the loss weight for all OOD unlabeled samples as zero so that the OOD samples are ignored in FixMatch training. The SimCLR-OOD model adds a SimCLR (Chen et al., 2020) loss term for OOD samples only in the model training. Please note that the two variant models have access to the ground truth ID/OOD labels for all unlabeled samples during training, while Ours needs to perform the ID/OOD classification in both training and testing. More details of the two variant models can be found in Sect. 5.4. As shown in Table 2, Mask-OOD outperforms the FixMatch baseline and demonstrates the benefit of OOD sample elimination in semi-supervised learning. More importantly, SimCLR-OOD works better than both FixMatch and Mask-OOD, which

Table 3 Justification of OOD samples' benefits in SSL

Method	DomainNet-Real		
(a)Normal case			
FixMatch (Sohn et al., 2020)	49.7 ± 2.5		
Mask-OOD	54.3 ± 0.7		
SimCLR-OOD	56.7 ± 0.4		
Clean	60.7 ± 0.8		
Datasets ID/OOD	CIFAR-100 10/90	SVHN S10/C100*	TinyImageNet 20/180
(b)Extreme case			
Labeled Only	47.3 ± 1.8	24.6 ± 2.4	32.2 ± 0.9
FixMatch	68.7 ± 1.5	43.4 ± 2.7	46.9 ± 0.4
Ours (Clustering)	**73.5 ± 1.3**	**50.2 ± 2.9**	**52.0 ± 0.9**

ID/OOD: the number of classes whose images are defined as ID and OOD samples. S10 / C100*: 10 ID classes are selected from SVHN and 100 OOD classes are selected from CIFAR-100

indicates that exploiting OOD samples properly can benefit the semi-supervised training and works better than simply filtering them out. Our method outperforms Mask-OOD and achieves comparable performance to SimCLR-OOD without access to ground-truth ID/OOD labels. The experiment results show that our method can exploit the OOD samples in open-set semi-supervised learning on larger benchmarks. More analysis can be found in Sect. 5.4.

5.4 Do OOD Samples Really Benefit SSL?

This section justifies the motivation of our method: if being "properly" used, OOD samples can benefit SSL. To verify this claim, we assume that *all unlabeled samples are perfectly identified as ID and OOD samples before training*. Based on this assumption, we propose two strategies to handle the OOD samples when training a FixMatch (Sohn et al., 2020) based SSL model:

- **Mask-OOD** masks all OOD samples by setting their weights to 0 in the FixMatch loss function.
- **SimCLR-OOD** adds a SimCLR loss term (Chen et al., 2020) for OOD samples in the FixMatch loss function.

Table 3 shows the results of Mask-OOD and SimCLR-OOD against the original FixMatch and "Clean" on the DomainNet-Real dataset. We use the same hyperparameters for all methods. Note that Mask-OOD is different from "Clean" as it does not remove the OOD samples and thus keeps the density of ID samples in mini-batches. It can be observed that: (i) Mask-OOD works better than the original FixMatch, which is consistent with the common belief that OOD is harmful to SSL. (ii) Mask-OOD works worse than SimCLR-OOD, which justifies our claim that *compared to filtering out OOD samples, exploiting them properly benefits SSL.* (iii) Mask-OOD works worse than "Clean", which indicates that the performance of SSL depends on the density of ID samples in a mini-batch. This motivates the use of our cascading pooling strategy. Please note that both Mask-OOD and SimCLR-OOD are conducted under a different setting to models in Table 1, as Mask-OOD and SimCLR-OOD have access to the ground truth ID/OOD label for all unlabeled data. In Table 1, the models do not know whether an unlabeled sample is ID or OOD but have to perform ID/OOD identification on their own.

Extreme Case Study. To further justify the motivation of our method, we test the performance of our method in an extreme case of open-set SSL where *all unlabeled samples are OOD*. To implement it, we remove all unlabeled ID samples from the training dataset. Note that we also remove the importance-based sampling method as it is useless in this scenario. Specifically, we compare our method (with clustering only) with FixMatch (Sohn et al., 2020) and its variant "Labeled Only"[2] on three datasets: CIFAR-100, SVHN and TinyImageNet. For CIFAR-100 and TinyImageNet, we set up the labeled ID samples and the unlabelled OOD samples within the same datasets. For SVHN, we set up the labeled ID samples from all its 10 classes and borrow the images from CIFAR-100 as the unlabeled OOD samples. As Table 3b shows, it can be concluded that Unlabeled OOD samples can still benefit SSL without unlabeled ID samples, which is justified by the observation that both Ours (Clustering) and FixMatch outperform "Labeled Only". This further justifies our motivation that OOD samples *DO* benefit SSL.

[2] In this case, "Clean" degenerates to "Labeled Only".

5.5 Ablation Study

This section studies the extent to which our proposed prototype-based clustering and identification algorithm and our importance sampling method contribute to the performance gains respectively. Specifically, we start from the original FixMatch (Sohn et al., 2020) and add our prototype-based clustering and identification algorithm, and our importance-based sampling method in turn. To further demonstrate the effectiveness of our method, we also tested several variants of our two components, including: *Clustering (Weak&Unlabeled-Only)*, which ignores the strongly-augmented samples and only clusters the weakly-augmented samples without the first term of Eq. 3 during training; *Clustering (Weak-Only)*, which ignores the strongly-augmented samples and only clusters the weakly-augmented samples during training with both Eq. 2 and Eq. 3; *Clustering* ($t_c = 0$), which sets the confidence threshold t_c in Eq 1 as 0 to remove the filtering strategy in clustering; *Refinement (Random)*, which randomly selects the ID samples identified in the clustering procedure for training; *Refinement (Importance)*, which removes the cascading sample pools and only uses importance-based sampling for training. All these methods are tested on two settings of the DomainNet-Real dataset with "Clean" as a reference. The experimental results are shown in Table 4. It can be observed that: (i) Our clustering and identification algorithm improves the performance over FixMatch by 2.2% and 2.8% respectively. The strong augmentation and the labeled sample clustering contribute to performance improvement with 0.8% and 0.4% on 10/50k, 0.9% and 0.6% on 20/50k respectively. (ii) Adding our importance-based sampling method can further improve the performance by 4.4% and 1.8% respectively (i.e. 6.6% and 4.6% higher than FixMatch). Note that "importance sampling only" is not a valid variant because our importance-based sampling method relies on the identification results and cannot be used independently. (iii) The filtering strategy is important for the performance of clustering. $t_c = 0$ reduces the accuracy by 1.6% and 1.5% on 10/50k and 20/50k respectively, because the clustering loss could cluster both high-confidence and low-confidence samples to the same prototype, and thus hinders the pseudo-labeling training of semi-supervised learning.

Table 5 Integrating our method in UDA (Xie et al., 2020) improves its performance on open-set SSL tasks (CIFAR-100)

Datasets	CIFAR-100	
ID/OOD	10/90	20/80
UDA (Xie et al., 2020)	38.9 ± 1.5	39.9 ± 2.1
+ Our Method	48.4 ± 1.1	43.1 ± 1.7
Clean(UDA)	67.9 ± 0.5	63.4 ± 0.9
FlexMatch (Zhang et al., 2021)	86.6 ± 0.3	80.9 ± 0.8
+ Our Method	88.0 ± 0.3	84.8 ± 0.4
Clean(FlexMatch)	88.1 ± 0.1	83.1 ± 0.1

To demonstrate that our method generalizes to other SSL methods, we integrate our method to UDA (Xie et al., 2020) and FlexMatch (Zhang et al., 2021) and test their performance on CIFAR-100 dataset (Table 5). It can be observed that our method improves the performance of UDA and FlexMatch under open-set settings by a significant margin.

5.6 Robustness Against ID/OOD Ratios

To demonstrate the robustness of our method against different ratios of ID/OOD samples in the training dataset, we test our method against the FixMatch (Sohn et al., 2020) baseline and its "Clean" variant on CIFAR-100 (Krizhevsky et al., 2009) dataset against four different ID/OOD ratios: 10/90, 20/80, 30/70 and 40/60. We report the average performance over three runs. As Fig. 5 shows, our method outperforms the FixMatch baseline in all four settings and achieves higher accuracy than the Clean model in three settings: 10/90, 20/80

Table 4 Ablation study

Datasets	DomainNet-Real	
Method	10/50k*	20/50k*
FixMatch (Sohn et al., 2020)	52.8 ± 2.9	49.7 ± 2.5
+ clustering (Weak&Unlabeled-Only)	53.8 ± 1.1	51.0 ± 0.4
+ clustering (Weak-Only)	54.2 ± 1.4	51.6 ± 0.5
+ clustering ($t_c = 0$)	53.4 ± 0.9	51.0 ± 0.8
+ clustering	55.0 ± 1.1	52.5 ± 0.7
+ refinement (Random)	57.2 ± 1.2	53.1 ± 1.3
+ refinement (Importance)	58.1 ± 0.4	53.7 ± 0.8
+ refinement (Ours)	59.4 ± 0.3	54.3 ± 1.2
Clean	63.5 ± 0.7	60.7 ± 0.8

$(\cdot)$/50k*: to balance the numbers of ID and OOD samples, we sample 50k images from classes other than $(\cdot)$ in DomainNet-Real as OOD samples

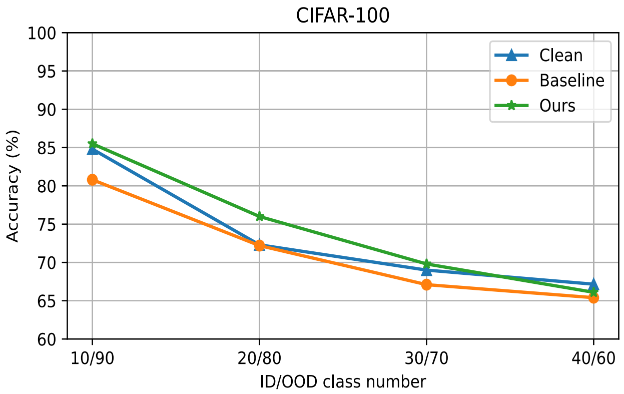


Fig. 5 The performance of FixMatch (Sohn et al., 2020), Clean and our method against four different ID/OOD ratios on CIFAR-100

and 30/70. Such a constant improvement justifies the robustness of our method against different settings of ID/OOD ratio. Note that the increase in the ID class number causes the degradation of performance in all three models.

5.7 Performance on ID/OOD Classification

Following previous studies (Saito et al., 2021), we also compare the performance of our method with those of previous open-set semi-supervised learning methods on ID/OOD classification. The experiments are conducted on CIFAR-100 with two different settings and the AUROC values of each method are shown in Table 6. As shown in the table, despite the imbalance between ID and OOD samples, our method achieves a significant improvement over previous methods. Please note that we use the output probabilities of the predicted class as ID probabilities to compute the AUROC value of FixMatch.

5.8 Justification of our Choice on the Number of Pools

To justify our choice of using a cascade of two pools in importance-based sampling, we investigate how the number of pools N_{pool} influences the performance of our method (Table 7) on DomainNet-Real with two ID/OOD settings. All other hyperparameters are kept the same across all experiments. It can be observed that: (i) using two pools achieves the best performance for both ID/OOD settings; (ii) when setting the number of pools to 0 or 1, the density of ID samples is not high enough and thus worsens the minibatch training; (iii) when using three or four pools, the density is improved but at the cost of filtering out too many unlabeled (ID) samples, which yields overfitting and also worsens the training. Thus, we use a cascade of two pools in our importance-based sampling implementation.

Table 6 Comparison of AUROC values for ID/OOD classification

Datasets	CIFAR-100	
ID/OOD	10/90	20/80
FixMatch	60.2 ± 0.2	57.0 ± 0.3
MTCF	70.6 ± 1.1	68.9 ± 1.4
OpenMatch	72.3 ± 0.8	71.5 ± 0.2
Our Method	79.6 ± 0.7	73.5 ± 0.5

Table 7 The performance of our method against different numbers of pool level N on DomainNet-Real

Dataset	DomainNet-Real	
ID/OOD	10/50k	20/50k
Ours($N_{pool} = 0$)	55.0 ± 1.1	52.5 ± 0.7
Ours($N_{pool} = 1$)	57.8 ± 0.7	53.2 ± 1.1
Ours($N_{pool} = 2$)	59.4 ± 0.3	54.3 ± 1.2
Ours($N_{pool} = 3$)	58.9 ± 0.6	52.3 ± 0.9
Ours($N_{pool} = 3$)	56.8 ± 1.0	50.9 ± 1.3

Table 8 The performance of our method against different N_{id} on DomainNet-Real

Dataset	DomainNet-Real	
ID/OOD	10/50k	20/50k
Ours($N_{id} = 5$)	59.6 ± 0.6	53.8 ± 0.9
Ours($N_{id} = 6$)	59.4 ± 0.3	54.3 ± 1.2
Ours($N_{id} = 7$)	58.7 ± 0.5	54.0 ± 1.0
Ours($N_{id} = 8$)	58.1 ± 0.5	53.1 ± 0.6

5.9 Threshold of ID/OOD Identification

The ID/OOD identification of our method selects N_{id} prototypes that are closest to the feature center of labeled samples as ID prototypes. To investigate the influence of N_{id} selection, we test our method on DomainNet-Real with two ID/OOD settings and the results are shown in Table 8. It can be observed that our method performs better than the baseline with different N_{id} and the best performance is achieved at 6 for 20/50k and 5 for 10/50k.

5.10 Number of Prototypes

To investigate how the number of prototypes influences the performance of our method, we test different choices of it on DomainNet-Real with two ID/OOD settings and show the results in Table 9. It can be observed that our method is insensitive to the number of prototypes and outperforms the baseline (FixMatch (Sohn et al., 2020)) in all experiments. Thus, we suggest to set the default value on DomainNet-Real as 30 (as used in this paper).

Table 9 The performance of our method against the number of prototypes K on DomainNet-Real

Dataset	DomainNet-Real	
ID/OOD	10/50k	20/50k
Ours($K = 20$)	58.7 ± 0.4	52.9 ± 1.3
Ours($K = 25$)	59.0 ± 0.6	53.7 ± 0.9
Ours($K = 30$)	59.4 ± 0.3	54.3 ± 1.2
Ours($K = 35$)	58.3 ± 0.8	53.4 ± 0.8
Ours($K = 40$)	57.1 ± 1.0	52.7 ± 0.5

5.11 Other Hyper-parameters Analysis

We further provide the performance of our method against different clustering threshold t_c, loss weight of L_c, the size of sample pools for each class N_P and the prototype initialization size L in Table 10. For each hyper-parameter, we test our method with four different settings on DomainNet-Real 20/50k. It can be observed that our method can achieve the best performance with $t_c = 0.70$, $w_{L_c} = 0.1$, $N_P = 300$ and $L = 250$. Besides, our method stably outperforms the FixMatch baseline in all settings.

5.12 Effectiveness of Cascading Pools

To further justify the effectiveness of our cascading pooling strategy, we plot the density of ID samples in our two cascaded ID sample pools (with per-class capacity 300 and 150 respectively) against training epochs when training our model with the DomainNet-Real 20/50k setting (Fig. 6). We also marked the percentage of ID samples in the raw unlabeled dataset as "Random Selection". It can be observed that: (i) Our two pools have much higher ID sample densities than Random Selection (approximately 10% and 20% respectively), which justifies the usefulness of our approach. (ii) Pool 2 has a much higher ID sample density than Pool 1 (approximately 10%), which indicates the effectiveness of our cascading pooling strategy.

Table 10 The performance of our method against different t_c, w_{L_c}, N_P and L (DomainNet-Real 20/50k)

t_c	0.55	0.65	0.70	0.75
Ours	52.0 ± 0.9	52.9 ± 1.3	54.3 ± 1.2	52.8 ± 1.2
w_{L_c}	0.05	0.1	0.15	0.2
Ours	53.3 ± 0.5	54.3 ± 1.2	53.9 ± 0.8	52.6 ± 0.6
N_P	250	300	350	400
Ours	52.8 ± 1.6	54.3 ± 1.2	53.9 ± 1.0	53.0 ± 0.9
L	150	200	250	300
Ours	53.7 ± 1.1	54.0 ± 0.9	54.3 ± 1.2	53.9 ± 0.8

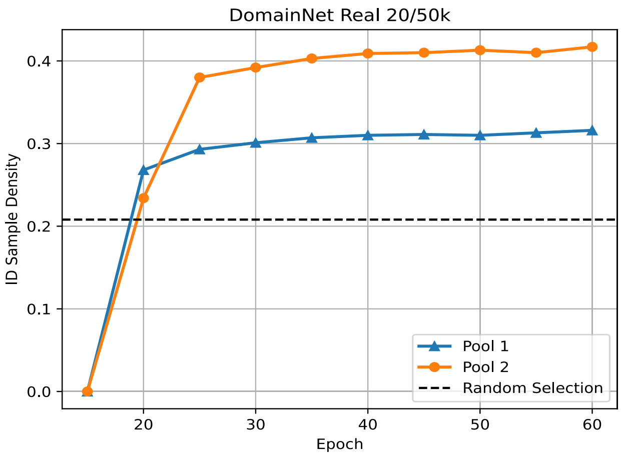


Fig. 6 Justification of ID sample refinement (DomainNet -Real 20/50k). The ID sample density of our ID sample pools is much higher than that of the raw unlabeled dataset (Random Selection)

5.13 Performance on Fine-Grained Classification

Fine-grained classification (Akata et al., 2015; Yang et al., 2018; Dubey et al., 2018; Syeda-Mahmood et al., 2020; Zhu et al., 2019) aims to distinguish between objects that previously belong to the same (coarse-level) class, e.g., species of birds. Recently, there have been some studies that apply open-set semi-supervised learning on fine-grained classification (Su et al., 2021), whose datasets contain both ID and OOD data. This is a more challenging task as samples in fine-grained classes (e.g., different brands of cars) have fewer discriminative features. In this section, we also verify the effectiveness of our method on fine-grained classification.

Datasets. Following previous studies (Su et al., 2021), we evaluate our method on two fine-grained datasets that exhibit a long-tailed distribution of classes and contain a large number of out-of-class images: Semi-Aves (from the semi-supervised challenge at FGVC7 workshop (Su & Maji, 2021)) and Semi-Fungi (from the FGVC fungi challenge (Brigit & Yin, 2018)). The OOD images of both datasets are those that do not belong to the classes of the labeled set. Between them, Semi-Aves contains 200 ID classes and 800 OOD classes, and 6K/27K/122K images in labeled set/ID unlabeled set/OOD unlabeled set, respectively. Semi-Fungi contains 200 ID classes and 1194 OOD classes, and 4K/13K/65K images in labeled set/ID unlabeled set/OOD unlabeled set, respectively. Following (Su et al., 2021), we use the labeled and unlabeled set (containing both ID and OOD samples) provided by these datasets and ResNet-50 (He et al., 2016) as the backbone network for evaluation. All samples are resized to a resolution of 224×224 in all experiments.

Comparison Setup. Following (Su et al., 2021), we compare our method to the following counterparts: (i) Supervised

Table 11 Results on Semi-Aves benchmark

Method	Top-1	Top-5
Supervised baseline	20.6 ± 0.4	41.7 ± 0.7
Pseudo-Label (Lee et al., 2013)	12.2 ± 0.8	31.9 ± 1.6
Curriculum Pseudo-Label (Cascante-Bonilla et al., 2021)	20.2 ± 0.5	41.0 ± 0.9
FixMatch (Sohn et al., 2020)	19.2 ± 0.2	42.6 ± 0.6
Self-Training	22.0 ± 0.5	43.3 ± 0.2
Ours	26.9 ± 0.5	48.4 ± 0.8
Supervised oracle	57.4 ± 0.3	79.2 ± 0.1

We experiment with six different SSL methods as well as supervised baselines. Results of other methods are copied from (Su et al., 2021)

Table 12 Results on Semi-Fungi benchmark

Method	Top-1	Top-5
Supervised baseline	31.0 ± 0.4	54.7 ± 0.8
Pseudo-Label (Lee et al., 2013)	15.2 ± 1.0	40.6 ± 1.2
Curriculum Pseudo-Label (Cascante-Bonilla et al., 2021)	30.8 ± 0.1	54.4 ± 0.3
FixMatch (Sohn et al., 2020)	25.2 ± 0.3	50.2 ± 0.8
Self-Training	32.5 ± 0.5	56.3 ± 0.3
Ours	34.4 ± 0.4	58.0 ± 0.8
Supervised oracle	60.2 ± 0.8	83.3 ± 0.9

We experiment with six different SSL methods as well as supervised baselines. Results of other methods are copied from (Su et al., 2021)

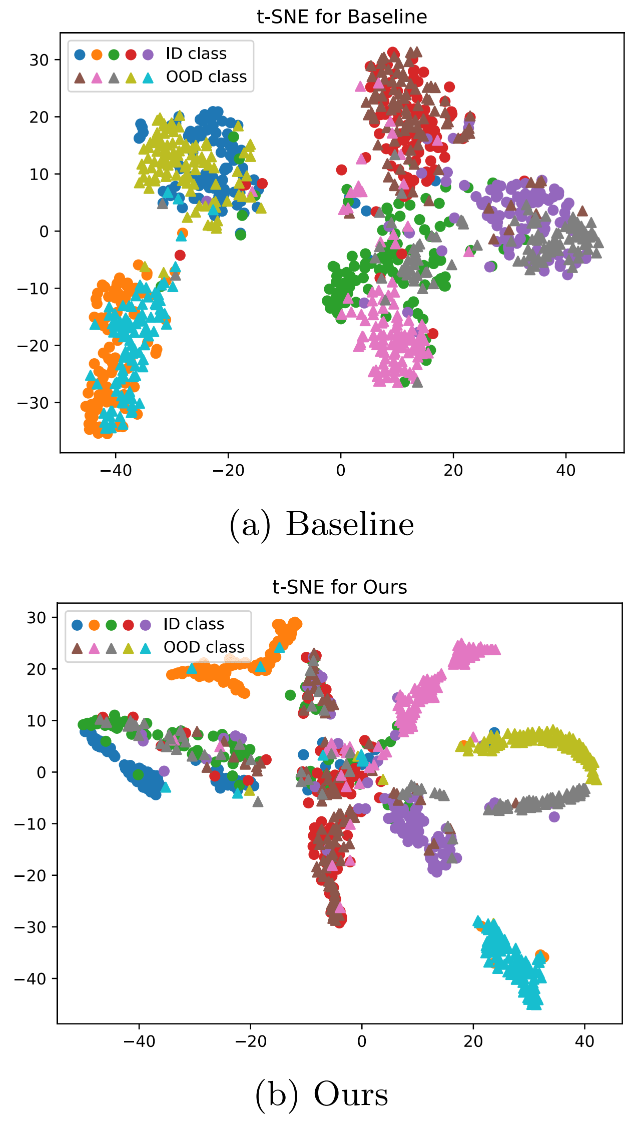
baseline, where the model is trained only with the labeled set; (ii) Pseudo-Labeling (Lee et al., 2013), which uses a base model's confident prediction on unlabeled images as pseudo-labels, and then trains a new model by sampling half of the batch from labeled data and half from pseudo-labeled data; (iii) Curriculum Pseudo-Labeling (Cascante-Bonilla et al., 2021), which repeats the following for 5 times: training a supervised model with labeled data, and expanding labeled data by including ({20, 40, 60, 80, 100}% of) the unlabeled data with the highest predictions. (iv) FixMatch (Sohn et al., 2020); (v) Self-Training, which first trains a teacher model with the labeled set, and then trains a student model with a scaled cross-entropy loss on the unlabeled data and a cross-entropy loss on the labeled data. (vi) Supervised Oracle, which trains the model with the labeled set and ID unlabeled set with ground-truth labels.

As shown in Tables 11 and 12, our method significantly outperforms all previous methods, which demonstrates the effectiveness of our method on fine-grained classification.

5.14 Visualization of ID/OOD Features

In this section, we visualize the features of unlabeled samples with and without our method using t-SNE (Van der Maaten & Hinton, 2008). We apply our method to CIFAR-10 to obtain the features as CIFAR-10 only contains 10 classes to be visualized. To better illustrate the difference and distribution of ID/OOD features, we select all 10 classes in CIFAR10 (Krizhevsky et al., 2009) rather than datasets with more categories in our experiment. We set the first 5 classes in CIFAR10 as ID and the other 5 classes as OOD. The feature visualization is shown in Fig. 7, including the visualization of both baseline (FixMatch) and our method. As shown in Fig. 7a, the baseline model can not separate the ID and OOD features well and thus confuses the OOD detector. Nevertheless, in Fig. 7b, our method can better cluster both ID/OOD features and thus preserves the difference between ID and OOD features in the feature level. Therefore, our method facilitates the training of the feature extractor and the ID/OOD classification in the importance-based sampling.

Figure 8 visualizes some image samples during the training on CIFAR-100 20/80. Three classes of images are shown: Boy, Bicycle, and Apple. For each class, we visualize the ID labeled data, ID unlabeled data, and In-Pool OOD unlabeled data. The ID unlabeled data and In-Pool unlabeled data are the samples that are stored in the cascading sample pool and utilized in the clustering loss. As shown in Fig. 8, our method tends to store the unlabeled samples in the pools with textures similar to the ID labeled samples. Besides, we further visualize Out-of-Pool OOD unlabeled data, which are the samples that are filtered by ID sample identification and not stored in the pools.



(a) Baseline

(b) Ours

Fig. 7 The t-SNE feature visualization on CIFAR-10. **a** Provides the feature visualization of FixMatch. **b** Shows the t-SNE results of our method. As shown in the figures, our method organizes both ID and OOD features and improves the feature extractor

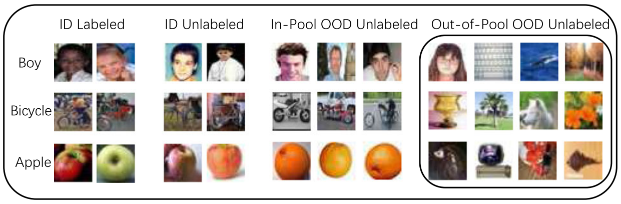


Fig. 8 The visualization of ID labeled data, ID unlabeled data, in-pool OOD unlabeled data, and out-of-pool OOD unlabeled data on CIFAR-100 during training. The OOD unlabeled data with similar textures to ID samples are more likely to be stored in sample pools and utilized for model training with the clustering loss

6 Conclusion

In this paper, we reveal that the proper use of OOD samples can benefit semi-supervised learning (SSL). Accordingly, we propose two techniques for open-set SSL: (i) a prototype-based clustering and identification algorithm and (ii) an importance-based sampling method. Our prototype-based clustering and identification algorithm clusters samples at the feature level and thus achieves better identification of ID and OOD samples by increasing their distances in-between. Addressing the sampling bias introduced by the ID/OOD identification process, we propose an importance-based sampling method that maintains a pyramid of sample pools containing samples that are important to SSL. We implemented our method on top of FixMatch (Sohn et al., 2020) and achieved state-of-the-art in open-set SSL on extensive public benchmarks.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (NO. 62322608), in part by the Shenzhen Science and Technology Program (NO. JCYJ20220530141211024), in part by the Guangdong Basic and Applied Basic Research Foundation under Grant NO. 2024A1515010255, and in part by the Open Project Program of the Key Laboratory of Artificial Intelligence for Perception and Understanding, Liaoning Province (AIPU, No. 20230003).

Availability of data and materials The data used in the paper is publicly available online.

Declarations

Conflict of interest The authors declare they have no Conflict of interest.

Ethical statements The datasets used in our work are officially shared by reliable research agencies, which guarantee that the collecting, processing, releasing, and using of data have gained the formal consent of participants. To protect privacy, all individuals are anonymized with simple identity numbers.

References

Akata, Z., Reed, S., Walter, D., Lee, H., & Schiele, B. (2015). Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2927–2936).

An, W., Tian, F., Zheng, Q., Ding, W., Wang, Q., & Chen, P. (2023). Generalized category discovery with decoupled prototypical network. *Proceedings of the AAAI Conference on Artificial Intelligence, 37*, 12527–12535.

Bachman, P., Alsharif, O., & Precup, D. (2014). Learning with pseudo-ensembles. *Advances in Neural Information Processing Systems, 27*

Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., & Raffel, C. (2019). Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International conference on learning representations*.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., & Raffel, C.A. (2019). Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, *32*

Brigit, S., & Yin, C. (2018). Fgvcx fungi classification challenge. Online.

Cascante-Bonilla, P., Tan, F., Qi, Y., & Ordonez, V. (2021). Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. *Proceedings of the AAAI Conference on Artificial Intelligence, 35*, 6912–6920.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607). PMLR.

Chen, Y., Zhu, X., Li, W., & Gong, S. (2020). Semi-supervised learning under class distribution mismatch. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*, 3569–3576.

Chow, C. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory, 16*(1), 41–46.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255). IEEE.

DeVries, T., & Taylor, G. W. (2018). Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865

Du, X., Gozum, G., Ming, Y., & Li, Y. (2022). Siren: Shaping representations for detecting out-of-distribution objects. In *Advances in neural information processing systems*.

Dubey, A., Gupta, O., Raskar, R., & Naik, N. (2018). Maximum-entropy fine grained classification. *Advances in Neural Information Processing Systems*, *31*.

Fan, Y., Kukleva, A., Dai, D., & Schiele, B. (2023). Ssb: Simple but strong baseline for boosting performance of open-set semi-supervised learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 16068–16078).

Ghoting, A., Parthasarathy, S., & Otey, M. E. (2008). Fast mining of distance-based outliers in high-dimensional datasets. *Data Mining and Knowledge Discovery, 16*(3), 349–364.

Grandvalet, Y., & Bengio, Y. (2004). Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems*, *17*.

Guo, L.-Z., Zhang, Z.-Y., Jiang, Y., Li, Y.-F., & Zhou, Z.-H. (2020). Safe deep semi-supervised learning for unseen-class unlabeled data. In *International conference on machine learning* (pp. 3897–3906). PMLR.

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (p. 9729–9738).

He, R., Han, Z., Lu, X., & Yin, Y. (2022). 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* (pp. 14585–14594).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

He, R., Han, Z., Yang, Y., & Yin, Y. (2022). Not all parameters should be treated equally: Deep safe semi-supervised learning under class distribution mismatch. *Proceedings of the AAAI Conference on Artificial Intelligence, 36*, 6874–6883.

Hendrycks, D., & Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International conference on learning representations*.

Hsu, Y.-C., Shen, Y., Jin, H., & Kira, Z. (2020). Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10951–10960).

Huang, J., Fang, C., Chen, W., Chai, Z., Wei, X., Wei, P., Lin, L., & Li, G. (2021). 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* (pp. 8310–8319).

Huang, Z., Yang, J., & Gong, C. (2022). They are not completely useless: Towards recycling transferable unlabeled data for class-mismatched semi-supervised learning. *IEEE Transactions on Multimedia*.

Krizhevsky, A., & Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

Laine, S., & Aila, T. (2017). Temporal ensembling for semi-supervised learning. In *International conference on learning representations*.

Lee, D.-H., *et al.* (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML* (vol. 3).

Lee, K., Lee, K., Lee, H., & Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, *31*.

Li, J., Zhou, P., Xiong, C., & Hoi, S. (2020). Prototypical contrastive learning of unsupervised representations. In *International conference on learning representations*.

Liang, S., Li, Y., & Srikant, R. (2018). Enhancing the reliability of out-of-distribution image detection in neural networks. In *International conference on learning representations*.

Liu, W., Wang, X., Owens, J., & Li, Y. (2020). Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems, 33*, 21464–21475.

Luo, H., Cheng, H., Gao, Y., Li, K., Zhang, M., Meng, F., Guo, X., Huang, F., & Sun, X. (2021). On the consistency training for open-set semi-supervised learning. arXiv preprint arXiv:2101.08237

Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, *9*(11).

Ming, Y., Cai, Z., Gu, J., Sun, Y., Li, W., & Li, Y. (2022). Delving into out-of-distribution detection with vision-language representations. In *Advances in neural information processing systems*.

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

Oliver, A., Odena, A., Raffel, C. A., Cubuk, E. D., & Goodfellow, I. (2018). Realistic evaluation of deep semi-supervised learning algorithms. *Advances in Neural Information Processing Systems*, *31*.

Park, S., Park, J., Shin, S.-J., & Moon, I.-C. (2018). Adversarial dropout for supervised and semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence* (vol. 32).

Park, J., Yun, S., Jeong, J., & Shin, J. (2022). Opencos: Contrastive semi-supervised learning for handling open-set unlabeled data. In *European conference on computer vision* (pp. 134–149). Springer

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., & Wang, B. (2019). Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1406–1415).

Pham, H., Dai, Z., Xie, Q., & Le, Q.V. (2021). Meta pseudo labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11557–11568).

Saito, K., Kim, D., & Saenko, K. (2021). Openmatch: Open-set consistency regularization for semi-supervised learning with outliers. In *Advances in neural information processing systems*.

Sajjadi, M., Javanmardi, M., & Tasdizen, T. (2016). Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in Neural Information Processing Systems*, *29*

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., & Li, C.-L. (2020). Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems, 33*, 596–608.

Su, J.-C., & Maji, S. (2021). The semi-supervised inaturalist-aves challenge at fgvc7 workshop. arXiv preprint arXiv:2103.06937

Su, J.-C., Cheng, Z., & Maji, S. (2021). A realistic evaluation of semi-supervised learning for fine-grained classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12966–12975).

Sun, Y., Guo, C., & Li, Y. (2021). React: Out-of-distribution detection with rectified activations. In *Advances in neural information processing systems*.

Syeda-Mahmood, T., Wong, K. C., Gur, Y., Wu, J. T., Jadhav, A., Kashyap, S., Karargyris, A., Pillai, A., Sharma, A., & Syed, A. B., *et al.* (2020). Chest x-ray report generation through fine-grained label learning. In *International conference on medical image computing and computer-assisted intervention* (pp. 561–571). Springer.

Tarvainen, A., & Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems*, *30*.

Vaze, S., Han, K., Vedaldi, A., & Zisserman, A. (2022). Generalized category discovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7492–7501).

Vincent, P., & Bengio, Y. (2003). Manifold parzen windows. *Advances in Neural Information Processing Systems, 849–856*.

Wager, S., Wang, S., & Liang, P. S. (2013). Dropout training as adaptive regularization. *Advances in Neural Information Processing Systems*, *26*

Wen, X., Zhao, B., & Qi, X. (2023). Parametric classification for generalized category discovery: A baseline study. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 16590–16600).

Winkens, J., Bunel, R., Roy, A. G., Stanforth, R., Natarajan, V., Ledsam, J. R., MacWilliams, P., Kohli, P., Karthikesalingam, A., & Kohl, S., et al. (2020). Contrastive training for improved out-of-distribution detection. arXiv preprint arXiv:2007.05566

Wu, Z., Xiong, Y., Yu, S. X., & Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3733–3742).

Xie, Q., Dai, Z., Hovy, E., Luong, T., & Le, Q. (2020). Unsupervised data augmentation for consistency training. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 33, pp. 6256–6268). Curran Associates Inc.

Yang, Z., Luo, T., Wang, D., Hu, Z., Gao, J., & Wang, L. (2018). Learning to navigate for fine-grained classification. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 420–435).

Yang, J., Wang, P., Zou, D., Zhou, Z., Ding, K., PENG, W., Wang, H., Chen, G., Li, B., & Sun, Y., et al.: Openood: Benchmarking generalized out-of-distribution detection. In *Thirty-sixth conference on neural information processing systems datasets and benchmarks track*.

Yu, Q., Ikami, D., Irie, G., & Aizawa, K. (2020). Multi-task curriculum framework for open-set semi-supervised learning. In *European conference on computer vision* (pp. 438–454). Springer.

Zagoruyko, S., & Komodakis, N. (2016). Wide residual networks. In *British machine vision conference 2016*. British Machine Vision Association.

Zhai, X., Oliver, A., Kolesnikov, A., & Beyer, L. (2019). S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1476–1485).

Zhang, S., Khan, S., Shen, Z., Naseer, M., Chen, G., & Khan, F.S. (2023). Promptcal: Contrastive affinity learning via auxiliary prompts for generalized novel category discovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, (pp. 3479–3488).

Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., & Shinozaki, T. (2021). Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems, 34*, 18408–18419.

Zhu, Y., Deng, X., & Newsam, S. (2019). Fine-grained land use classification at the city scale using ground-level images. *IEEE Transactions on Multimedia, 21*(7), 1825–1838.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.