Semi-Supervised Learning via Weight-aware Distillation under Class Distribution Mismatch

Pan Du[1,2], Suyun Zhao[1,2,*], Zisen Sheng[1, 2], Cuiping Li[1,2], Hong Chen[1,2]
Key Lab of Data Engineering and Knowledge Engineering of MOE Renmin University of China[1]
Renmin University of China, Beijing, China[2]
{du_pan,shengzisen}@163.com, {zhaosuyun, zisen, licuiping, chong}@ruc.edu.cn

Abstract

Semi-Supervised Learning (SSL) under class distribution mismatch aims to tackle a challenging problem wherein unlabeled data contain lots of unknown categories unseen in the labeled ones. In such mismatch scenarios, traditional SSL suffers severe performance damage due to the harmful invasion of the instances with unknown categories into the target classifier. In this study, by strict mathematical reasoning, we reveal that the SSL error under class distribution mismatch is composed of pseudo-labeling error and invasion error, both of which jointly bound the SSL population risk. To alleviate the SSL error, we propose a robust SSL framework called Weight-Aware Distillation (WAD) that, by weights, selectively transfers knowledge beneficial to the target task from unsupervised contrastive representation to the target classifier. Specifically, WAD captures adaptive weights and high-quality pseudo-labels to target instances by exploring point mutual information (PMI) in representation space to maximize the role of unlabeled data and filter unknown categories. Theoretically, we prove that WAD has a tight upper bound of population risk under class distribution mismatch. Experimentally, extensive results demonstrate that WAD outperforms five state-of-the-art SSL approaches and one standard baseline on two benchmark datasets, CIFAR10 and CIFAR100, and an artificial cross-dataset. The code is available at `https://github.com/RUC-DWBI-ML/research/tree/main/WAD-master`.

1. Introduction

Deep neural networks (DNNs) have achieved remarkable success in fully-supervised learning tasks. However, sufficient labeled data are usually unavailable in real applications due to the expensive annotation cost or even domain-specific knowledge required [8, 11, 12, 13]. Semi-supervised learning (SSL), as a powerful weakly-supervised technique, provides an effective way to improve DNNs by exploiting massive unlabeled data, and then it weakens the demand for human annotation [9, 14, 24, 34]. Generally, traditional SSL approaches assume that the labeled and unlabeled instances share the same class distribution, i.e., they come from identical categories. However, in real scenarios, this assumption hardly holds as unlabeled data inevitably contains lots of categories unseen in labeled ones. For instance, if unlabeled data are collected from the internet using keywords "cat" and "dog" (target categories), they may contain instances unrelated to these categories, such as "deer," "horse," or "airplane"(unknown categories), as shown in Figure 1. Similar scenarios occur in medical diagnoses [11, 15] and house annotations of remote-sensing images [12, 13]. SSL in such mismatch scenarios is called SSL under class distribution mismatch [12, 15].

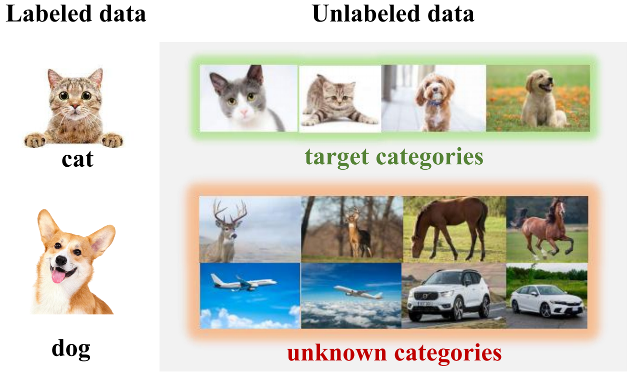


Figure 1. Example of class distribution mismatch. The unlabeled data contains categories that are unseen in labeled ones.

Under class distribution mismatch, some SSL approaches [8, 11, 15, 19, 37] have been proposed. Usually, most of them exploit pseudo-labeling or consistency regularization to expand the labeled pool, as well as filter instances with unknown categories by weights, just as shown in Figure 2. UASD [11] and T2T [19] filter out the instances with unknown categories by leveraging a hard weight, i.e., a

[*] Corresponding Author

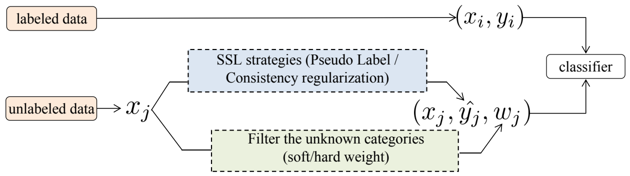

labeled data
(x_i, y_i)
classifier
SSL strategies (Pseudo Label / Consistency regularization)
unlabeled data
x_j
$(x_j, \hat{y}_j, w_j)$
Filter the unknown categories (soft/hard weight)

Figure 2. The paradigm of SSL under class distribution mismatch.

threshold, on the accumulated network's output or the out-of-distribution score. Although these two approaches reduce the invasion of unknown categories, it is inevitable to keep off amounts of unlabeled instances with target categories. Instead of hard weights, Guo et al. [15] assign a soft weight to the unlabeled instances according to the consistent empirical risk loss. In such case, many instances with unknown categories tend to have consistent outputs and get high weights, just as shown in Appendix 4.3, and then they may invade the target classifier and impair its performance.

Moreover, the existing SSL approaches with consistency regularization and pseudo-labeling heavily rely on the performance of the target classifier. Both [15] and [19] annotate pseudo labels by leveraging the prediction of the target classifier in training. Once the target classifier trained on limited labeled instances is biased by some instances with unknown categories, the subsequently updated target classifier may allow more unknown instances to invade. Accordingly, it is promising to propose a novel SSL approach that captures pseudo labels from representations produced by all available data rather than an immature classifier.

In this study, by strict theoretical analyses, we decouple the SSL error under class distribution mismatch into pseudo-labeling error and invasion error (seen in Subsection 3.2). According to this discovery, a robust SSL framework called weight-aware distillation (WAD) is then proposed to distill pseudo labels and weights from the representation space to the target classifier. Unlike the conventional distillation approaches [7, 17, 28] that simply train the student model using the prediction probability of the teacher model, WAD is a weight-aware distillation framework that adapts to mismatch problems. Specifically, we learn the representations from labeled and unlabeled data by unsupervised contrastive coding, as the teacher model. Then WAD captures adaptive weights as well as high-quality pseudo labels from the teacher model by leveraging point mutual information(PMI), and thus, the target classifier could selectively utilize the instances from target categories while filtering the ones with unknown categories.

Our main contributions are listed as follows.

i) We theoretically analyze the population risk in an SSL manner and reveal that the SSL error under class distribution mismatch is jointly controlled by pseudo-labeling error and invasion error.

ii) We propose a distillation-based SSL framework, WAD, that captures weights as well as pseudo labels from robust representations to the target classifier to filter unknown categories and make full use of targeted unlabeled instances as well.

iii) Theoretically, we verify that the population risk of WAD is tightly bounded. Experimentally, WAD outperforms five state-of-the-art SSL approaches and one standard baseline on several datasets.

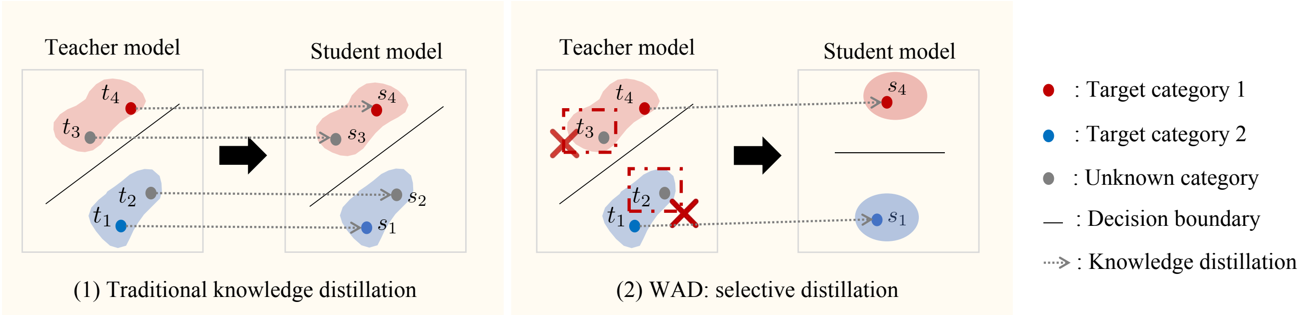
2. Related Work

This section reviews the SSL approaches under class distribution match and mismatch. For contrastive learning, please refer to Appendix 1.

Semi-Supervised Learning. The traditional SSL strategies include entropy minimization, consistency regularization, and pseudo-label. Entropy minimization [14] incorporates unlabeled data in supervised learning by minimizing the entropy of the unlabeled instance's prediction. The consistency regularization [24, 31, 34] techniques mainly make the prediction on two views of one instance consistent. Π-Model [31] focuses on reducing the distance of prediction between one instance and its stochastic perturbation. Unlike the Π-Model, temporal ensembling [24] adopts the ensemble of predictions as the target to achieve more stable performance, while Virtual Adversarial Training (VAT) [27] explores adversarial disturbances of the unlabeled instances on the prediction of the target classifier. Pseudo-Labeled based approaches [3, 4, 26, 33] annotate some unlabeled instances with pseudo labels to expand the labeled data. By leveraging the class probability of the unlabeled data, a pseudo-labeling method is proposed [26]. Furthermore, FixMatch [33] uses the weakly augmented unlabeled instances to create a pseudo label and enforce consistent prediction against its strong augmented version.

These traditional SSL approaches perform well when the class distribution is matched, but they suffer severe performance degradation under class distribution mismatch.

Semi-Supervised Learning under Class Distribution Mismatch. To tackle class distribution mismatch, several studies [11, 15, 19, 37] adopt the traditional SSL strategies with the assistance of soft or hard weights. UASD [11] leverages a threshold to the accumulated network's output to eliminate the instances with unknown categories, followed by pseudo-labeling highly confident ones. Similarly, T2T [19] adopts a hard weight on the out-of-distribution score to conduct filtering and leverages consistency constraints to expand the labeled pool. Furtherly, CCSSL [37] filters out unknown instances by taking both hard and soft weights into consideration. These approaches with hard weights may eliminate too many instances from target categories. Instead of hard weights, DS^{3}L [15] assigns soft



(1) Traditional knowledge distillation

(2) WAD: selective distillation

Figure 3. The selective distillation in WAD. Traditional distillation methods force the output of the student model (s_i) to align with that of the teacher model (t_i), resulting in a wrong decision boundary. However, WAD selectively distills benefit knowledge to student and filter the negatives, such as s_2 and s_3, by weights to rectify the decision boundary and solve the problem of class distribution mismatch.

weights to unlabeled instances according to the consistent empirical risk loss. However, SSL with pseudo labeling or consistency regularization heavily rely on the performance of the target classifier, and thus they are susceptible to being invaded by instances with unknown categories.

Additionally, a model-level approach [40] is proposed by modifying batch normalization to counter the unknown categories. Also, ORCA [8], a novelty detection approach, leverages uncertainty-based adaptive margins to circumvent the bias caused by the mismatched distribution.

Knowledge Distillation. Knowledge distillation aims to transfer knowledge from a big model (teacher model) to a smaller one (student model) [35]. It is widely applied to two distinct fields: model compression and knowledge transfer. Model compression is training a small student model to mimic the big teacher model or the ensemble of models. Buciluǎ et al. [7] compress the ensembles of the neural networks into a single one. While the approaches based on transfer knowledge concentrate more on effectively transferring and are mainly divided into logits-based and representation-based distillation [39]. The logits-based distillation approaches usually train the student model by leveraging the output of the teacher model as the soft label [35]. Ba et al. [1] propose to push the logits, i.e., the output before the softmax function, of the shallow neural network to mimic the ones from a deep neural network. Furtherly, Hinton et al. [17] suggest training a student model to match the combination of the softmax distribution of the teacher model and ground truth. Representation-based approaches enable the student model to learn information from the intermediate layers [35]. Kim et al. [21] propose transferring the attention map from the teacher to the student. Park et al. [30] introduce a novel approach that transfers the mutual relationship of the instances learned from the teacher to the student, similar to our intention.

However, these approaches mentioned above aim to transfer as much information as possible to the student model and ignore the unknown instances under class distribution mismatch, which may severely hurt the training of the student model. Unlike the conventional approaches, WAD is a weight-aware distillation framework that selectively transfers the knowledge to the student model, as shown in Figure 3, to fully use the beneficial knowledge and filter the unknown ones by weights. Specifically, WAD distills high-quality pseudo labels to the instances with target categories and filters the instances with unknown categories by assigning them tiny weights.

3. Method

In this section, we propose WAD, an SSL framework under class distribution mismatch. Concretely, Subsection 3.1 introduces the problem statement, followed by analyses of the SSL error in Subsection 3.2. Subsection 3.3 subsequently presents WAD. Finally, theoretical studies of WAD are conducted in Subsection 3.4.

3.1. Problem Statement

In this study, we investigate the K classification problem in an SSL manner wherein limited labeled data $\mathcal{D}_l = \{(x_{i,l}, y_{i,l})\}_{i=1}^{m}$ and massive unlabeled instances $\mathcal{D}_u = \{x_{i,u}\}_{i=1}^{n}$ are accessible, $x_{i,l} \in \mathcal{X}$, $y_{i,l} \in \mathcal{Y}$, $\mathcal{Y} = \{1, ..., K\}$ and $m \ll n$. Under class distribution mismatch, the unlabeled instances are not guaranteed to belong to the K target categories in $\mathcal{Y}$.

3.2. Population Risk Analysis

To make full use of the unlabeled data, we assign a pseudo label to each unlabeled instance, denoted as $\hat{y}$, and then build the target classifier, $h_{\hat{\mathcal{T}}} : \mathcal{X} \rightarrow \mathcal{Y}$, to map the given instance to one of the known categories in $\mathcal{Y}$, where $\hat{\mathcal{T}} = \{x_{i,l}, y_{i,l}\}_{i=1}^{m} \cup \{x_{i,u}, \hat{y}_{i,u}\}_{i=1}^{n}$, $\hat{y}_{i,u} \in \mathcal{Y}$. Here, $\hat{\mathcal{T}}$ indicates the instances in hand, that is, labeled instances and unlabeled instances assigned with pseudo labels. Then, the population risk [32] of the target classifier learned from both labeled and unlabeled data with the pseudo label ($\hat{\mathcal{T}}$) is controlled by the generalization gap, training error, and SSL error, as shown in Eq.1. The generalization gap is the gap between the population risk and the average prediction loss across all instances with target categories (T). Note that T contains all the accessible instances with target categories, including labeled and unlabeled. And every instance in T is assumed with ground truth labels in ideal. The training error is the average empirical loss across $\hat{T}$. The SSL error is the gap between the average empirical loss across the instances with target categories (T) and the average empirical loss across both labeled data and unlabeled ones with pseudo labels ($\hat{T}$). We depict the relations among these sets in Figure 4.

$$\begin{aligned}
&\mathbb{E}_{(\boldsymbol{x},y)\sim D}[l(\boldsymbol{x},y;h_{\hat{T}})] \\
&\le \underbrace{\left|\mathbb{E}_{(\boldsymbol{x},y)\sim D}[l(\boldsymbol{x},y;h_{\hat{T}})] - \frac{1}{|T|}\sum_{(\boldsymbol{x},y)\in T} l(\boldsymbol{x},y;h_{\hat{T}})\right|}_{\textbf{generalization gap}} \\
&+ \underbrace{\left|\frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in \hat{T}} l(\boldsymbol{x},y;h_{\hat{T}})\right|}_{\textbf{training error}} \qquad (1) \\
&+ \underbrace{\left|\frac{1}{|T|}\sum_{(\boldsymbol{x},y)\in T} l(\boldsymbol{x},y;h_{\hat{T}}) - \frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in \hat{T}} l(\boldsymbol{x},y;h_{\hat{T}})\right|}_{\textbf{SSL error}},
\end{aligned}$$

where $\mathcal{D}$ is the data distribution of the instances that belong to target categories in the realistic world, i.e., $\mathcal{D} = \mathcal{X} \times \mathcal{Y}$. $l(\cdot,\cdot;h_{\hat{T}}) : \mathcal{X} \times \mathcal{Y} \to \mathcal{R}$ denotes the loss function of the classifier $h_{\hat{T}}$ learned from $\hat{T}$.

Theoretical analyses [36] have confirmed that the generalization gap of DNNs can be bounded, and empirical evidence suggests that the training error of DNNs can be reduced almost to zero [32]. Thus, the essential component concerning population risk is the SSL error. Under class distribution mismatch, in addition to the wrongly annotated instances with target categories, the ones with unknown categories also contribute to the SSL error as they invade the training of the target classifier as outliers. Accordingly, we decouple the SSL error into pseudo-labeling and invasion error, as shown in Eq.2. For a detailed derivation process, please refer to Appendix 5.2.

$$\begin{aligned}
&\left|\frac{1}{|T|}\sum_{(\boldsymbol{x},y)\in T} l(\boldsymbol{x},y;h_{\hat{T}}) - \frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in \hat{T}} l(\boldsymbol{x},y;h_{\hat{T}})\right| \\
&\le \underbrace{\left|\frac{1}{|T|}\sum_{(\boldsymbol{x},y)\in T} l(\boldsymbol{x},y;h_{\hat{T}}) - \frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in \hat{T}\backslash U} l(\boldsymbol{x},y;h_{\hat{T}})\right|}_{\textbf{Pseudo}-\textbf{labeling error}} \\
&+ \underbrace{\left|\frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in U} l(\boldsymbol{x},y;h_{\hat{T}})\right|}_{\textbf{Invasion error}} \qquad (2)
\end{aligned}$$

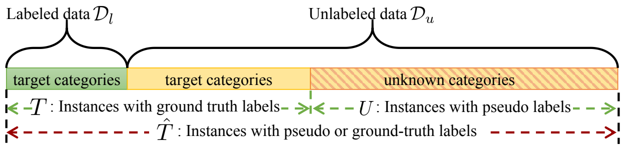

Labeled data $\mathcal{D}_l$
Unlabeled data $\mathcal{D}_u$
target categories
target categories
unknown categories
T : Instances with ground truth labels
U : Instances with pseudo labels
$\hat{T}$: Instances with pseudo or ground-truth labels

Figure 4. The relations among data sets $\mathcal{D}_l$, $\mathcal{D}_u$, T, U, $\hat{T}$. Note that $\hat{T} \neq T \cup U$ due to the instances with target categories in $\mathcal{D}_u$ are assigned with pseudo labels while not the ground truth ones in T.

where U indicates the unlabeled instances with unknown categories and $\hat{T} = \hat{T}\backslash U \cup U$.

In Eq.2, the pseudo-labeling error is contributed by the wrongly annotated instances with target categories, as it is the gap of the average empirical loss caused by the inconsistency of the ground truth and pseudo labels. Thus, the quality of pseudo-labels assigned to unlabeled instances within the target distribution determines this error, and accurate pseudo-labeling may alleviate it. By contrast, the invasion error is the average empirical loss across the instances with unknown categories that is caused by the negative effect of those untargeted instances. By Eq.1& Eq.2, we find that the population risk of the target model is jointly controlled by pseudo-labeling error and invasion error. Accordingly, to mitigate the SSL error, we need to filter those instances with unknown categories and accurately annotate the unlabeled instances with target categories as well.

3.3. Weight-aware Distillation Framework

With the aim of mitigating the pseudo-labeling and invasion errors, we design an SSL framework named WAD, which delivers the knowledge of pseudo labels and weights from robust representations to the target classifier.

3.3.1 Pseudo Label Learning

Most existing SSL approaches produce pseudo labels by leveraging an immature target classifier, which cause catastrophic error once invaded by some instances with unknown categories, just as discussed in Section 1. To solve this problem, we distill the pseudo labels from a representation space (Teacher model) which is learned from all labeled and unlabeled instances by contrastive learning in an unsupervised manner and then transfer it to the target classifier (Student model). The teacher model could produce closely aligned representations for instances from the same categories and maximize the mutual information among them [2, 18, 20, 25].

Denoted the labeled and unlabeled representations learned by the teacher model, ϕ, as $\mathcal{Z}_l = \{z_{j,l,k}\}_{j=1}^{m}$ and $\mathcal{Z}_u = \{z_{i,u}\}_{i=1}^{n}$, respectively, where $k \in \mathcal{Y}$. Inspired by the characteristic of contrastive learning, one effective approach for building the pseudo-label of an unlabeled in-

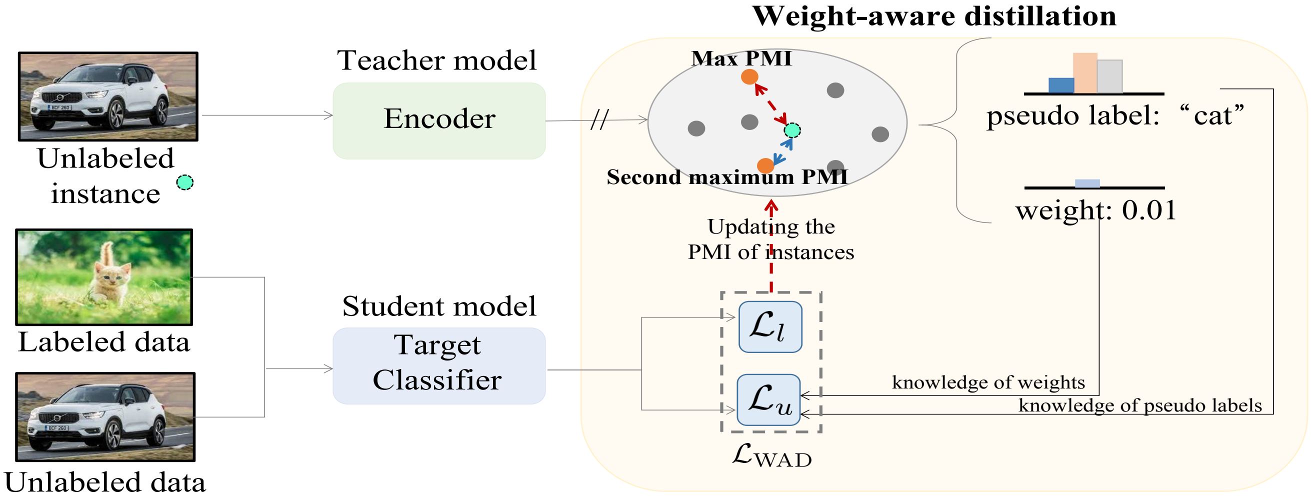


Figure 5. Illustration of WAD. The pseudo labels and weights determined by point mutual information (PMI) in robust representations of the teacher model are used to participate in training the target classifier. Then, some reliable instances selected according to $\mathcal{L}_{\text{WAD}}$ are regarded as labeled ones to update the knowledge gradually. “●” and “●” indicates the labeled and unlabeled instance respectively. “//” means stop gradient.

stance is to identify the labeled instance with the highest PMI and then assign the label of it to unlabeled ones. The PMI between the unlabeled and labeled representation is formulated as Eq.3.

$$\text{PMI}(z_{i,u}, z_{j,l,k}) = log\left[\frac{p(z_{j,l,k}|z_{i,u})}{p(z_{j,l,k})}\right] \tag{3}$$

Although the conditional and marginals distributions, i.e., $p(z_{j,l,k}|z_{i,u})$ and $p(z_{j,l,k})$, cannot be directly evaluated, we prove that PMI is proportional to the inner product in Appendix 2, as described in Eq.4.

$$f(z_{i,u}, z_{j,l,k}) \propto \text{PMI}(z_{i,u}, z_{j,l,k}). \tag{4}$$

where $f = cos(\cdot,\cdot)$, $\|z_i\| = 1$, and $\propto$ stands for “proportional to”.

Therefore, the pseudo label is formulated as Eq.5.

$$\hat{y}_{i,u} = \arg\max_k f(z_{i,u}, z_{j,l,k}) \tag{5}$$

Consequently, the class label of the labeled instance with the maximum PMI is assigned to the unlabeled one. The Eq.5 can precisely capture the PMI from the representation space to produce high-quality pseudo labels and then mitigate pseudo-labeling error.

3.3.2 Unknown Categories Filtering

To mitigate the invasion error, the instances with unknown categories should be filtered out. Following the Subsubsection 3.3.1, a higher PMI between the labeled and unlabeled instance suggests a stronger association or similarity between the two instances, further indicating a higher likelihood of the unlabeled instance belonging to the same class distribution as the labeled one. However, some hard instances that have similar PMI between two target categories, i.e., laid on the decision boundary of two target categories, may introduce incorrect pseudo labels and hurt the performance of the target classifier. Hence, we also propose a ratio among the first and second maximum PMI to evaluate the confidence of the pseudo labels. Then, the weight is defined as Eq.6 to avoid the negative effect caused by the wrong labels and unknown categories.

$$\boldsymbol{w}_{i,u} = g_1\left(\widetilde{p}_{i,u}\right) \times g_2\left(1 - \frac{\widetilde{q}_{i,u}}{\widetilde{p}_{i,u}}\right), \tag{6}$$

wherein,

$$\widetilde{p}_{i,u} = \max_j f(z_{i,u}, z_{j,l,k})$$
$$\widetilde{q}_{i,u} = \max_{v,k\neq\hat{y}_{i,u}} f(z_{i,u}, z_{v,l,k})$$

In Eq.6, $g_1(\cdot)$ and $g_2(\cdot)$ can be interpreted as any monotonically increasing functions. The former in Eq.6 aims to estimate the likelihood of the unlabeled instance belonging to target categories. The higher this item, the more chances of the instance in the target class distribution are. The latter, $g_2(\cdot)$, penalizes instances whose labels are ambiguous between the nearest and second-nearest target categories. The lower this item, the larger probability of incorrect pseudo labels is. As shown in Figure 6, the weight could filter instances with unknown categories and those incorrectly annotated ones with target categories, while the ones from target categories with high-quality pseudo labels are encouraged. Thus, by weight, WAD selectively distills the knowledge beneficial to the target classifier from the teacher model, and the invasion error is then mitigated.

3.3.3 Weight-aware Knowledge Distillation

Weight-aware knowledge distillation loss. The knowledge of pseudo labels and weights captured from robust representations is applied in the distillation process. In each feed-forward process, pseudo labels and weights are aggregated to the target classifier, as shown in Figure 5. Then, we propose the weight-aware knowledge distillation loss, including the traditional supervised loss $\mathcal{L}_l$ in labeled data and weight-aware supervised loss $\mathcal{L}_u$ in unlabeled data as Eq.7.

$$\mathcal{L}_{\text{WAD}} = \mathcal{L}_l + \mathcal{L}_u, \tag{7}$$

wherein,

$$\mathcal{L}_l = \frac{1}{|\mathcal{D}_l|} \sum_{(x_{i,l}, y_{i,l}) \in \mathcal{D}_l} \ell(h(x_{i,l};\theta), y_{i,l})$$
$$\mathcal{L}_u = \frac{1}{|\mathcal{D}_u|} \sum_{x_{i,u} \in \mathcal{D}_u} \boldsymbol{w}_{i,u} \ell(h(x_{i,u};\theta), \hat{y}_{i,u})$$

The traditional supervised loss $\mathcal{L}_l$ aims to minimize the distance between the predicted probability and the ground truth label. While the weight-aware supervised loss $\mathcal{L}_u$ mainly focuses on selectively transferring the beneficial knowledge from the teacher model to the student model by weights to mitigate the negative effect from unknown categories and improve the target classifier as well. Moreover, $\mathcal{L}_{\text{WAD}}$ is the loss function that is adopted to train the target classifier $h_{\hat{T}}$ mentioned in Eq.1. Consequently, WAD leverages the pseudo labels and weights to mitigate the pseudo-labeling and invasion errors, following alleviating the SSL error, which has been proved in Subsection 3.4.

Knowledge-update in Training. The knowledge of pseudo labels and weights may be biased as the labeled data is limited. Accordingly, after several forward iterations, we progressively add some reliable instances to labeled data. Because the feedback from the target classifier, i.e., loss, is highly related to the weights and reflects the training error, we consider the reliability according to it. Then, the criterion for updating is formulated as Eq.8.

$$c_{i,u} = \ell(h(x_{i,u};\theta_t), \hat{y}_{i,u}) \tag{8}$$

where $\ell(\cdot,\cdot)$ is the cross-entropy function, and θ_t is the parameters of the target classifier in the current iteration.

The reliability of $x_{i,u}$ is enhanced when $c_{i,u}$ takes a lower value. Then, WAD leverages Eq.8 to identify the top $\alpha\%$ reliable instances from the unlabeled data and puts them in the labeled data while removing them from the unlabeled data. Moreover, we adopt the polynomial decay [5] to dynamically adjust α to prevent the gradually increased negative effect from unknown categories with the iteration. The details are shown in Appendix 3. A visualization of the number of selected reliable instances with target categories is also provided in Appendix 4.4. Consequently, the pseudo labels and weights are updated in the subsequent distillation steps, as shown in Figure 5, with the aim of optimizing the target classifier. Finally, the schematic diagram and algorithm process is presented in Figure 6 and Algorithm 1, respectively.

3.4. Theoretical Studies

This subsection provides the theoretical studies about the WAD's SSL error, as shown in Theorem 1. Detailed proof of Theorem 1 is given in Appendix 5.

Theorem 1 *Given $|T|$ instances that i.i.d. sampled from $\mathcal{D}$ as $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|T|}$, $|U|$ instances that is not i.i.d with $\mathcal{D}$, and $\hat{T} = \{(x,\hat{y})|(x,y) \in T \cup U, \hat{y} \in \mathcal{Y}\}$ where $\mathcal{D} = \mathcal{X} \times \mathcal{Y}$. Assume the loss function $l(\cdot, y; h_{\hat{T}})$ is λ^l-Lipschitz continuous for all $y, h_{\hat{T}}$ and bounded by H, the regression function is λ^μ-Lipschitz continuous, training error $l(\boldsymbol{x}, y; h_{\hat{T}}) = 0$, $\forall(\boldsymbol{x}, y) \in \hat{T}$. $\overline{\boldsymbol{w}}$ indicates the average of weights, and ξ is the maximum PMI which determines the pseudo label, with the probability of at least $1-\gamma$,*

$$\begin{aligned}
&\left|\frac{1}{|T|}\sum_{(\boldsymbol{x},y)\in T} l(\boldsymbol{x}, y; h_{\hat{T}}) - \frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in \hat{T}\setminus U} l(\boldsymbol{x}, y; h_{\hat{T}})\right| \\
&+ \left|\frac{1}{|\hat{T}|}\sum_{(\boldsymbol{x},y)\in U} l(\boldsymbol{x}, y; h_{\hat{T}})\right| \qquad (9)\\
&\leq \sqrt{4-4\xi}(\lambda^l + \lambda^\mu HK) + \frac{\overline{\boldsymbol{w}}|U|H}{|\hat{T}|} + \sqrt{\frac{2H^2 log(1/\gamma)}{|T|}}.
\end{aligned}$$

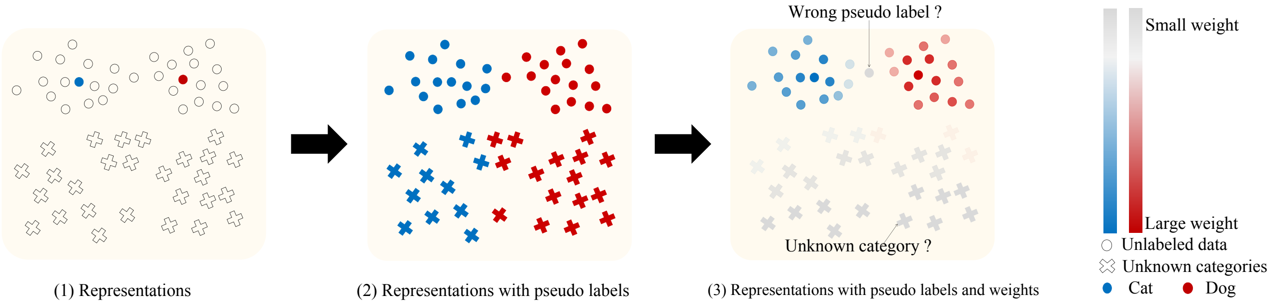
From Theorem 1, we find that the smaller $\overline{\boldsymbol{w}}$ and the larger ξ, the tighter the bound in the SSL error is. Specifically, just as verified in Appendix 5, the pseudo-labeling error bounded by $\sqrt{4-4\xi}(\lambda^l + \lambda^\mu HK) + \sqrt{\frac{2H^2 log(1/\gamma)}{|T|}}$ and the invasion error bounded by $\frac{\overline{\boldsymbol{w}}|U|H}{|\hat{T}|}$ can be reduced by minimizing the weights w of unlabeled instances with unknown categories and maximizing the confidence of pseudo labels, just as WAD does. Thus, WAD's SSL error has a tight upper bound.

4. Experiments

Subsection 4.2 presents the comparison results between WAD and five state-of-the-art SSL approaches, as well as one standard baseline. Furthermore, an ablation experiment is conducted in Subsection 4.3, while sensitivity analyses and visualization are carried out in Subsection 4.4 and Subsection 4.5, respectively. For more experiments, please refer to Appendix 4.2 & 4.5.

4.1. Experimental Setups

Datasets. Our experiments are conducted on two benchmark datasets, CIFAR10 [23] and CIFAR100 [23], as well as an artificial cross-dataset that comprises subsamples



(1) Representations

(2) Representations with pseudo labels

(3) Representations with pseudo labels and weights

Figure 6. The schematic diagram of how WAD works. First, each unlabeled instance is assigned a pseudo label according to the max PMI with labeled data in (1), as shown in (2). Then, the instances with wrong labels and unknown categories are assigned a tiny weight to avoid the negative effect, as shown in (3).

```
Algorithm 1: Weight-aware distillation framework
Input: Labeled data: $\mathcal{D}_l$, Unlabeled data: $\mathcal{D}_u$, Max
       iterations: $N$, Initial value of $\alpha$: $\alpha_0$, Set of
       update steps: $G$.
Output: target classifier $h_{\hat{T}}$.
Embedding: $\mathcal{Z}_l \leftarrow \phi(\mathcal{D}_l)$, $\mathcal{Z}_u \leftarrow \phi(\mathcal{D}_u)$.
Initialize model parameter;
for $t = 0$ to $N-1$ do
    $\forall \boldsymbol{x}_{i,u} \in \mathcal{D}_u$, distill the knowledge about pseudo
      label, $\hat{y}_{i,u}$, using Eq.5;
    $\forall \boldsymbol{x}_{i,u} \in \mathcal{D}_u$, distill the knowledge about weight,
      $\boldsymbol{w}_{i,u}$, using Eq.6;
    $\alpha$ = polynomial_decay($\alpha_0$);
    Using Eq.7 training the target classifier;
    if $t \in G$ then
        $C = \emptyset$;
        Calculate the reliability of each unlabeled
          instance using Eq.8.
        Ascending the instances according to
          reliability and add the top $\alpha$% instances
          with pseudo labels to $C$.
        $\mathcal{D}_l = \mathcal{D}_l \cup C$, $\mathcal{D}_u = \mathcal{D}_u \backslash \{x|(x, y) \in C\}$.
end
Return target classifier $h_{\hat{T}}$.
```

from CIFAR10, CIFAR100, Flowers [29], Food-101 [6], and Places-365 [41]. The CIFAR10 and CIFAR100 datasets consist of 50,000 training and 10,000 testing images of 10 and 100 categories, respectively. The cross-dataset contains 138,000 unlabeled instances from 674 categories. All images from the datasets are resized to 32×32. For further details, please refer to Appendix 4.1.

Settings. i) The proportion of the instances with unknown categories in unlabeled data, named as mismatch proportion, are set as 20%, 40%, 60%, and 80% in this work. For instance, the unlabeled data has a 60% mismatch proportion with 4,000 instances with target categories and 6,000 instances with unknown categories. ii) Randomly sampled 8% instances from the training dataset that belong to target categories are regarded as labeled data. The remaining 92% of instances with target categories and some instances with unknown categories are composed of unlabeled data according to the mismatch proportion.

Details. The teacher model is with a Resnet-18 [16] backbone and is trained using SimCLR [10]. And we maintain consistency with SimCLR in all implementation details. The target classifier is a WideResnet-28-2 network [38] with input size 32 × 32, following Huang et al. [19]. Both the encoder and target classifier are trained from scratch. We train the target classifier using the Adam optimizer [22] with a learning rate of 5×10^{-4}. Furthermore, the epochs and batch size are set as 100 and 32, respectively. The augmentations include random horizontal flipping, random translation by up to 2 pixels, and Gaussian input noise with a standard deviation of 0.15 is used in the training of the target classifier as same as Guo et al. [15]. Moreover, we apply global contrast normalization and ZCA normalized, which is widely used in the pretreatment [15, 11], on CIFAR10. For simplicity, the functions $g_1(\cdot)$ and $g_2(\cdot)$ act as identical mapping with no additional constraints. The initial value of α is set as 0.1 and decayed five times until it reached 0. It remains the same across all experiments unless otherwise specified. Finally, the approaches on each dataset run three times, and the mean accuracy and standard deviation are reported; the best one is highlighted in bold.

Baselines. WAD is compared to five state-of-the-art approaches, including DS^{3}L [15], T2T [19], CCSSL [37], UASD [11] and ORCA [8], as well as one baseline model that only trains labeled data. Moreover, T2T and ORCA are performed without pretraining tasks for fairness, indicated by "T2T **w**\o pre." and "ORCA **w\o** pre." .

4.2. Experimental Results

This subsection presents the experimental results of the classification tasks performed on CIFAR10, CIFAR100,

	CIFAR10				CIFAR100			
Method	20%	40%	60%	80%	20%	40%	60%	80%
Baseline	94.33±0.45	94.33±0.45	94.33±0.45	94.33±0.45	36.98±1.79	36.98±1.79	36.98±1.79	36.98±1.79
DS^{3}L	91.82±1.89	91.38±1.73	92.47±0.25	90.82±1.50	23.92±2.78	24.92±4.41	26.20±4.29	24.55±3.67
UASD	95.02±0.77	95.03±0.77	93.87±0.13	93.37±0.35	39.85±0.35	37.55±2.24	36.03±0.73	29.87±2.07
CCSSL	86.08±0.12	84.00±0.17	83.13±0.19	81.15±0.25	41.72 ±0.85	41.20±0.58	40.60 ±0.22	39.67±0.31
T2T	-	-	-	-	43.70±0.50	42.82±0.45	40.12±0.71	37.35±1.10
T2T**w\o** pre.	-	-	-	-	39.40±0.36	36.78±0.16	36.65 ±1.09	34.62±1.68
ORCA	95.40±0.74	94.13±1.16	94.35±0.67	93.82±0.93	29.50±0.25	31.12±0.71	31.18±0.40	31.65±1.86
ORCA **w\o** pre.	93.32±0.99	92.55±2.02	92.37±0.90	89.65±6.95	22.13±1.33	23.98±0.79	23.37±1.14	22.98±0.53
WAD	**98.43±0.14**	**97.88±0.33**	**97.90±0.20**	**97.77±0.33**	**51.65±2.86**	**50.00±1.43**	**46.88±0.20**	**45.45±1.73**

Table 1. Experimental results on CIFAR10 and CIFAR100 under different mismatch proportions.

and a cross-dataset. For CIFAR10, we designated two categories as the target and eight as unknown, while twenty classes are considered as target categories and eighty categories as unknown in CIFAR100. Moreover, we constructed a cross-dataset integrated with five datasets to evaluate WAD in the case that the unlabeled data contains massive unknown categories. Specifically, six classes from CIFAR10 were assigned as target categories, and 668 categories from four external datasets are unknown. The experimental results conducted on CIFAR10, CIFAR100, and the cross-dataset are presented in Table 1 and Table 2.

Results on CIFAR10 and CIFAR100. From Table 1, we have four findings as follows. i) WAD outperforms all compared methods on CIFAR10 and CIFAR100 with different mismatch proportions, demonstrating its remarkable performance. ii) WAD retains stable performance improvement under different mismatch proportions, exhibiting further improvement of 4.1%, 3.55%, 2.91%, and 3.44% for mismatch proportions of 20%, 40%, 60%, and 80% on CIFAR10. This highlights that WAD can achieve robust performance even under a high mismatch proportion. iii) The accuracies of DS^{3}L on CIFAR10 and CIFAR100 are lower than baseline, as ORCA does. This is because it weights the instances according to consistent empirical risk loss, resulting in the invasion of many unknown categories in training, as shown in Appendix 4.3. iv) In CIFAR100, WAD surpasses baseline 8.47% for 80% mismatch proportion. This demonstrates that WAD is still effective when the unlabeled data contains large unknown categories. Therefore, WAD achieves outstanding performance on datasets with different mismatch proportions and exhibits excellent robustness to the scale of unknown categories. Notably, T2T can not apply to the binary classification task, and the accuracy is not reported here.

Results on cross-dataset. Further, we investigate the limits of WAD's tolerance for unknown categories and then perform the experiments on an artificial cross-dataset containing 668 unknown categories from four datasets. From Table 2, we observe that WAD still maintains an improvement compared to the baseline. Obviously, the other compared methods were lower than the baseline. This indicates that WAD could boost the performance even on a dataset that contains massive instances with unknown categories.

	Cross-dataset			
Method	20%	40%	60%	80%
Baseline	66.83±1.37	66.83±1.37	66.83±1.37	66.83±1.37
DS^{3}L	50.02±6.69	50.69±5.26	49.03±5.93	51.46±6.99
UASD	61.18±0.29	57.02±0.58	54.70±2.25	45.67±1.72
CCSSL	64.83±0.27	65.15±0.56	64.16±0.58	64.16±0.45
T2T	66.56±2.80	65.08±0.76	63.76±0.53	62.83±0.77
T2T **w\o** pre.	64.44±0.15	62.47±0.79	62.23±0.75	61.42 ±0.65
ORCA	65.53±0.85	65.51±1.25	66.44±0.80	66.46±1.28
ORCA **w\o** pre.	65.37±0.78	63.63±0.64	64.42±0.53	66.34±1.05
WAD	**67.13±0.59**	**67.20±1.65**	**67.80±0.07**	**67.88±0.37**

Table 2. Experimental results on cross-dataset under different mismatch proportions.

4.3. Ablation Studies

We conducted ablation studies on the CIFAR10 dataset using different models: "+Pse." (trained with labeled data and unlabeled instances with pseudo labels), "+Pse.&W." (trained with pseudo labels and fix weights), and the WAD model (trained with all components). We also examined the impact of the weight function and explored alternative choices for $g(\cdot)$ through identical mappings, $g_i(\cdot)$, and the transformation $\tilde{g}_i(\cdot) = exp(\cdot)$. Results are presented in Table 3, and w\o $g_i(\cdot)$ means removing $g_i(\cdot)$ from Eq.6.

Effects of pseudo labels. From Table 3, we observe that compared with the baseline, 2.72% and 1.52% accuracy improvement can be obtained by leveraging the unlabeled instances with pseudo labels, under 20% and 80% mismatch proportion, respectively. This indicates that the pseudo labels are beneficial to improving performance.

Effects of weights. According to Table 3, we observe two findings about weights. i) Training by leveraging both pseudo labels and weights exhibits the comparable performance to that without weights under 20% mismatch proportion. This is because there are fewer instances with unknown categories under 20%. Then, the model training with fixed weights will result in a sub-optimal model compared to explicit labels. ii) The weights improve the accuracy by 0.99% over without it, under 40% mismatch proportion, while the gap decreases with the mismatch proportion increasing. This demonstrates that the weights are effective in filtering the instances with unknown categories. And the performance degradation is because the absence of the knowledge-update makes the algorithm fail to prevent the increased negative effect from unknown categories.

Effects of knowledge-update. We have two findings according to Table 3. i) The accuracy with knowledge-update surpasses the one only leveraging pseudo labels and weights, and the gap between them reaches 1.77% under 80% mismatch proportion. This indicates that knowledge-update plays important roles in WAD. ii) WAD, training with all components, shows its outstanding performance compared to the ones removing other parts. This demonstrates that the aggregation of all the proposed parts could achieve significant improvement.

Effects of each part of Eq.6. From the Table 3, we have the following two findings. i) both "w\o $g_1(\cdot)$" and "w\o $g_2(\cdot)$" are worse than WAD, illustrating their equal importance for WAD. ii) Assigning the same mappings to $g_1(\cdot)$ and $g_2(\cdot)$ yields better performance, as the same mappings share the same scales.

Setting	20%	40%	60%	80%
Baseline	94.33±0.45	94.33±0.40	94.33±0.4	94.33±0.45
+Pse.	97.05±0.48	95.98±0.75	96.65±0.35	95.85±0.88
+Pse.&W	96.62±0.47	96.97±0.78	97.22±0.38	96.00±0.48
w\o $g_1(\cdot)$	97.85±0.57	96.98±0.11	94.38±1.52	94.38±0.74
w\o $g_2(\cdot)$	97.98±0.53	96.85±0.14	94.58±0.46	95.85±0.35
$\tilde{g_1}(\cdot) \times g_2(\cdot)$	97.50±0.07	97.15±0.71	94.95±0.14	94.48±0.46
$g_1(\cdot) \times \tilde{g_2}(\cdot)$	96.60±0.57	96.93±0.04	95.65±0.21	95.65±0.07
WAD	**98.43±0.14**	**97.88±0.33**	**97.90±0.20**	**97.77±0.33**

Table 3. Ablation Studies under different mismatch proportions.

4.4. Sensitivity Analysis

This subsection investigates the influence of parameter α, which controls how many instances with high reliability will be added to labeled data. Hence, we vary the initial value of α and evaluate WAD's performance on CIFAR10. The results are reported in Table 4. We find that WAD depicts the comparable performance with different values of α, although lower values of α achieve slightly better performance with 20% and 40% mismatch proportions. This indicates that WAD is not sensitive to α because the selected instances may have higher similarities. Thus, WAD can achieve a robust performance for a wide range of α but not too large, preventing the invasion of unknown categories.

4.5. Visualization

To comprehend how WAD works, we visualize pseudo labels assigned to unlabeled instances with target categories alongside the ground truths of labeled ones in different colors, as depicted in the left part of Figure 7. We observe that instances with target categories are separated into two clusters and follow the same distribution as labeled instances. Additionally, the weight distribution, shown on the right side of Figure 7, depicts that WAD assigns smaller weights to unknown categories and larger ones to target ones, making it feasible to filter out harmful unknown categories and to distill useful information from target ones.

Setting	20%	40%	60%	80%
$\alpha = 0.05$	**98.62±0.28**	**98.12±0.39**	97.63±0.03	97.63±0.19
$\alpha = 0.10$	98.43±0.14	97.88±0.33	97.90±0.20	**97.77±0.33**
$\alpha = 0.15$	98.35±0.25	97.73±0.15	**97.93±0.08**	97.42±0.63

Table 4. Sensitivity analysis under varied mismatch proportions.

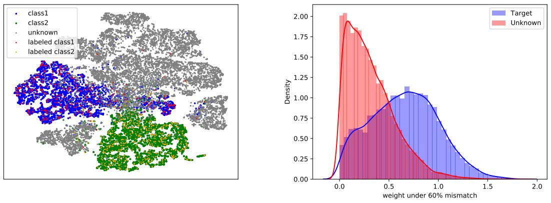


Figure 7. Visualization of pseudo labels and weights on CIFAR10 under 60% class distribution mismatch.

5. Conclusions

To tackle class distribution mismatch in an SSL manner, we theoretically reveal that the SSL error is composed of pseudo-labeling error and invasion error under mismatch scenarios. Then, a distillation-based SSL framework, WAD, is proposed to transfer knowledge, such as pseudo labels and weights, from the representations to the target model. Theoretical analyses verify that the population risk of WAD is tightly bounded. Extensive experiments on two benchmark datasets and a cross-dataset demonstrate the superiority of WAD.

In the near future, we would like to investigate whether some instances from unknown categories are beneficial to target task and how to utilize them if so.

6. Acknowledgement

This work is supported by the National Key Research & Develop Plan(2018YFB1004401), National Natural Science Foundation of China(62276270,62072460, 62172424), Beijing Natural Science Foundation(4212022), Fundamental Research Funds for Central University, and the Research Funds of Renmin University of China. It is also partially supported by the Opening Fund of Hebei Key Laboratory of Machine Learning and Computational Intelligence.

References

[1] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? *Advances in neural information processing systems*, 27, 2014. 3

[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019. 4

[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 *International Conference on Learning Representations*, 2019. 2

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019. 2

[5] Alexander Borichev and Yuri Tomilov. Optimal polynomial decay of functions and operator semigroups. *Mathematische Annalen*, 347(2):455–478, 2010. 6

[6] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014. 7

[7] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006. 2, 3

[8] Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning. In *International Conference on Learning Representations*, 2021. 1, 3, 7

[9] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009. 1

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 7

[11] 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*, volume 34, pages 3569–3576, 2020. 1, 2, 7

[12] Pan Du, Hui Chen, Suyun Zhao, Shuwen Chai, Hong Chen, and Cuiping Li. Contrastive active learning under class distribution mismatch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13, 2022. 1

[13] Pan Du, Suyun Zhao, Hui Chen, Shuwen Chai, Hong Chen, and Cuiping Li. Contrastive coding for active learning under class distribution mismatch. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8927–8936, October 2021. 1

[14] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004. 1, 2

[15] 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*, pages 3897–3906. PMLR, 2020. 1, 2, 7

[16] 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*, pages 770–778, 2016. 7

[17] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *In NIPS Deep Learning and Representation Learning Workshop*, 2015. 2, 3

[18] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2018. 4

[19] 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*, pages 8310–8319, 2021. 1, 2, 7

[20] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2021. 4

[21] Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 736–751, 2018. 3

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. 7

[23] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[24] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 1, 2

[25] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 2020. 4

[26] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 2

[27] 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*, 41(8):1979–1993, 2018. 2

[28] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In *International Conference on Machine Learning*, pages 4743–4751. PMLR, 2019. 2

[29] M-E Nilsback and Andrew Zisserman. A visual vocabulary for flower classification. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1447–1454. IEEE, 2006. 7

[30] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019. 3

[31] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29, 2016. 2

[32] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. 3, 4

[33] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020. 2

[34] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 1, 2

[35] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[36] Huan Xu and Shie Mannor. Robustness and generalization. *Machine learning*, 86(3):391–423, 2012. 4

[37] Fan Yang, Kai Wu, Shuyi Zhang, Guannan Jiang, Yong Liu, Feng Zheng, Wei Zhang, Chengjie Wang, and Long Zeng. Class-aware contrastive semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14421–14430, 2022. 1, 2, 7

[38] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*. British Machine Vision Association, 2016. 7

[39] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11953–11962, 2022. 3

[40] Xujiang Zhao, Killamsetty Krishnateja, Rishabh Iyer, and Feng Chen. Robust semi-supervised learning with out of distribution data. *arXiv preprint arXiv:2010.03658*, 2020. 3

[41] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017. 7