Pre-Trained Vision-Language Models as Noisy Partial Annotators

Qian-Wei Wang[1,2], **Yuqiu Xie**[1], **Letian Zhang**[1], **Zimo Liu**[2], **Shu-Tao Xia**[1,2]

[1]Tsinghua Shenzhen International Graduate School, Tsinghua University
[2]Pengcheng Laboratory
{wanggw21, jieyq22, zlt23}@mails.tsinghua.edu.cn, liuzm@pcl.ac.cn, xiast@sz.tsinghua.edu.cn

Abstract

In noisy partial label learning, each training sample is associated with a set of candidate labels, and the ground-truth label may be contained within this set. With the emergence of powerful pre-trained vision-language models, e.g. CLIP, it is natural to consider using these models to automatically label training samples instead of relying on laborious manual annotation. In this paper, we investigate the pipeline of learning with CLIP annotated noisy partial labels and propose a novel collaborative consistency regularization method, in which we simultaneously train two neural networks, which collaboratively purify training labels for each other, called Co-Pseudo-Labeling, and perform consistency regularization between label and representation levels. For instance-dependent noise that embodies the underlying patterns of the pre-trained model, our method employs multiple mechanisms to avoid overfitting to noisy annotations, effectively mines information from potentially noisy sample set while iteratively optimizing both representations and pseudo-labels during the training process. Comparison experiments with various kinds of annotations and weakly supervised methods, as well as other pre-trained model application methods demonstrates the effectiveness of method and the feasibility of incorporating weakly supervised learning into the distillation of pre-trained models.

Code — https://github.com/Portugas0807/Co-Reg

Introduction

Partial label learning (PLL) (Lv et al. 2020; Zhang and Yu 2015; Lyu et al. 2021; Wang et al. 2022a; Wu, Wang, and Zhang 2022), a type of weakly supervised learning (WSL), deals with the classification problem where each training example is associated with multiple candidate labels, called partial label, among which only one label is the ground-truth. Due to the fundamental assumption that the ground-truth label must be within the candidate label set, which cannot be satisfied in many scenarios and thus affects the practical applicability of PLL, recent years have seen a growing focus on noisy partial label learning (NPLL) (Qiao et al. 2023; Shi et al. 2023b,a; Wang et al. 2022b). In NPLL, the aforementioned assumption is relaxed, allowing the ground-truth labels of some noisy samples to appear outside their candidate label sets.

Previous research on NPLL typically assumes that the partial annotations of training dataset originate from manually labelling manners. With the emergence of powerful pre-trained vision-language models, e.g. CLIP (Radford et al. 2021), which learns from massive "image-natural language" pairs and can generalize to unseen tasks by leveraging textual categorical description, it is natural to consider using these models to automatically label training samples instead of relying on laborious manual annotation.

In this paper, we investigate the pipeline of employing pre-trained vision-language model CLIP to annotate images of downstream tasks with partial labels, and then training a specialized model using these partial labels. Here, each prompt template corresponds to a classification result of a sample, and the classification results from multiple prompt templates collectively form the candidate label set. Compared to simulated settings such as random noise (also called symmetric noise), which are typically experimented on by previous NPLL research, training on datasets annotated by CLIP becomes significantly more challenging. This is because the noise here is instance-dependent and follows the underlying patterns of the pre-trained model. Specifically, even if a dataset contains a large amount of random noise, since the model cannot learn any effective patterns from it, the algorithm can easily identify that those noisy labels conflict with the current model's output and consequently correct them by assigning pseudo-labels. However, when dealing with noise from a pre-trained model, the algorithm can easily misinterpret noisy labels as high-confidence true labels, since they may align with the patterns learned from the pre-trained model's annotations.

To address this issue, we propose a novel **Co**llaborative consistency **Reg**ularization (Co-Reg) method. Our method simultaneously trains two neural networks, which collaboratively purify training labels for each other, called Co-Pseudo-Labeling, and performs consistency regularization between label and representation levels. In Co-Pseudo-Labeling, we train two neural networks simultaneously and each network divides the pre-trained model's annotated partial labels into valid and noisy, which are then provided to the other network for training. This approach effectively

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

reduces the confirmation bias that can arise from mimicking the pre-trained model. For samples identified as having noisy partial labels, we regard them as unlabeled samples and aggregate the outputs from multiple data-augmented variants predicted by both neural networks for consistency regularization. Utilizing, rather than discarding, the unlabeled split is crucial for the model to achieve performance improvements, since it largely corrects the errors made by the original pre-trained model on the downstream task. When performing self-training, our method not only uses the obtained pseudo-label as the training objective for the output class distribution but also maintains a representation prototype for each class and then calculates the prototypical similarity distribution of the sample representation and aligns it with the pseudo-label for additional calibration. Furthermore, our method introduces a noise-tolerant supervised contrastive learning module, aimed at enabling the model to learn more discriminative representations, which enhances the advantage of the specialized small model over the general pre-trained model in specific tasks.

Our main contributions can be summarized as:

- To the best of our knowledge, we are the first to investigate the NPLL problem in the context of learning with pre-trained model annotation and propose a novel collaborative consistency regularization algorithm.
- We conduct comparative experiments with various kinds of annotations and WSL methods, as well as other pre-trained model application methods including zero-shot learning, knowledge distillation, and prompt learning.
- We discuss the comparison of introducing WSL with other pre-trained model application paradigms and demonstrate the feasibility of WSL and NPLL on image classification tasks. This inspiration can be further extended to various types of tasks, pre-trained models, and WSL scenarios.

Related Work

Noisy Partial Label Learning

PLL (Zhang and Yu 2015; Feng and An 2018; Wang, Li, and Zhou 2019; Wang et al. 2023) is an important branch of weakly-supervised learning, in which each training sample is associated with multiple candidate labels, among which only one is valid. The main difficulty of this problem is recognizing the ground-truth label among other associated false-positive candidates, which is called "disambiguation" of the partial label. In recent years, data-augmentation-based consistency regularization has been widely applied to PLL, leading to the emergence of a series of impressive methods (Wu, Wang, and Zhang 2022; Wang et al. 2022a; Xia et al. 2023). They achieved performances with only minor differences compared to fully supervised training counterpart, even when the training labels contained a large proportion of false-positives.

However, traditional PLL imposes an overly strict assumption that the ground-truth labels of all samples must be included within the their candidate label sets, which can hardly be met in scenarios such as crowdsourcing and learning with pre-trained model annotation. As a result, there has been a growing tendency to study a more practical extension of PLL, known as NPLL (Lian et al. 2022; Wang et al. 2022b; Xu et al. 2024; Shi et al. 2023b,a; Qiao et al. 2023). NPLL allows the existence of noises of partial labels, i.e. the ground-truth label is not any of the candidate labels. Previous methods usually design specific mechanisms to handle noisy partial samples simultaneously with partial label disambiguation. Some methods (Qiao et al. 2023; Lian et al. 2022; Shi et al. 2023a) incorporate non-candidate labels with high predicted probabilities into the candidate label sets during training. Shi et al. (2023b) divide the samples based on whether they contain noise. Xu et al. (Xu et al. 2024) assign a certain probability to non-candidate labels in the training objective.

However, previous NPLL methods primarily address simulated settings such as random noise and show unsatisfactory performances in the scenario of learning with pre-trained model annotations. Xu et al. (Xu et al. 2021) study instance-dependent partial labels by using a neural network trained on ground-truth labels to obtain the probability of flipping false-positive candidate labels. However, this scenario remains a simulated crowdsourcing setting and has not been extended to NPLL.

Applying CLIP to Downstream Tasks

With the invention of effective multi-modal deep frameworks and training algorithms, as well as significant improvements in computing power, pre-trained vision-language models (Li et al. 2022; Bao et al. 2022; Achiam et al. 2023) have demonstrated impressive capabilities on a wide range of tasks. Taking CLIP (Contrastive Language-Image Pre-Training) (Radford et al. 2021) as an example, it models unified representations of images and natural language by learning massive "image-text" pairs. During pre-training, semantically related pairs of images and texts are encoded by the image encoder and the text encoder to obtain aligned representations. Because CLIP learns from natural language descriptions, it can leverage the rich contextual information provided by language to understand and categorize new images without having been explicitly trained on them. When presented with a new downstream task, CLIP can interpret the category descriptions which are then matched by the encoded input images in the shared embedding space.

There are two main issues when directly applying CLIP to downstream image classification tasks: 1. For tasks where the input image domain differs significantly from the general domain, CLIP often fails to accurately match the input images with the corresponding category descriptions, resulting in poor performances; 2. In scenarios with limited computational resources, the overhead of executing large model inference may be unsustainable. To enhance the performance and efficiency of pre-trained vision-language models, several adaptation and distillation techniques have been explored.

One approach is prompt learning (Zhou et al. 2022b,a; Khattak et al. 2023), which involves fine-tuning the input textual or visual embeddings of vision-language models. This approach can be seen as an extension of manual prompt

engineering, which designing appropriate prompts for better aligning downstream images and textual descriptions by domain experts. Since prompt learning only fine-tunes the embedding vectors while keeping the text and image encoder frozen, it requires only few-shot training samples.

Similarly, Adapter (Houlsby et al. 2019; Gao et al. 2024) has been proposed as a lightweight adaptation mechanism for pre-trained models. Adapters allow the model to learn task-specific information while retaining the majority of its pre-trained parameters. This approach offers a more efficient alternative to fully fine-tuning, enabling quick adaptation to new tasks without the need of large amount of data and significant computational overhead.

Knowledge distillation is a technique where a smaller, student model is trained to replicate the behavior of a larger, teacher model. This approach aims to transfer knowledge from the large, often cumbersome models to more compact and efficient versions, maintaining high performance while reducing computational requirements. (Hinton, Vinyals, and Dean 2015) laid the groundwork for this technique, demonstrating that distillation can significantly compress models without substantial loss in accuracy.

CLIP Annotated Partial Labels

This section introduces how we use CLIP to annotate downstream image datasets. We use a collection of prompt templates, denoted as $\{\mathcal{T}_1(\cdot), \mathcal{T}_2(\cdot) \dots \mathcal{T}_d(\cdot)\}$, and combine them with the class names of images to form the textual input. The template here is like: "a photo of a {}.", where {} is replaced with class names. We denote the class names as $\{n_1, n_2, \dots, n_C\}$, and denote the combination of j-th class and i-th template as $\mathcal{T}_i(n_j)$.

For each template $\mathcal{T}_i(\cdot)$, we combine it with all class names and then take them as the input of CLIP text encoder and obtain C textual representations $\{t_1, t_2, \dots, t_C\}$, where $t_j = \text{TextEncoder}(\mathcal{T}_i(n_j))$. Input the training image into the CLIP Image Encoder and obtain image representation r. Then, we can predict the probabilities of the image belonging to different classes under this prompt template as $p_i = \text{softmax}(rt_1, rt_2, \dots, rt_C)$.

We then obtain the numerical label from each predicted probabilities $\hat{p}_i = \arg\max p_i$, and deem each prompt template's label as a candidate and form the partial label $Y = (y_1, y_2, \dots, y_C) \in \{0, 1\}^C$, in which $y_j = 1$ if $j \in S$ and $y_j = 0$ if $j \notin S$, and S denotes the candidate label set formed by numerical label $\hat{p}_i$.

In experiments, we also compare the methods that annotate noisy single-labels by averaging the predicted probabilities p_i of all prompt templates and then train on these labels with corresponding algorithms. We found that annotating partial labels achieves better results, especially under extreme circumstances when most prompt templates fail to provide satisfactory predictions. And at this time, as long as one prompt template makes a correct prediction, the prediction will be included in the candidate label set and the difficult of the algorithm to recognize it as the correct label with the help of consistency regularization is greatly decreased. This is very helpful when the characteristic of downstream task is unknown and prompt engineering can hardly be performed.

Methodology

Supervised Training

To avoid overfitting to the pre-trained model's outputs, as is common in traditional unsupervised knowledge distillation, our method applies supervised training on the pre-trained model's annotations for only a few warm-up epochs. We adopt the partial cross-entropy loss as the supervised loss, as well as the negative entropy to prevent from overfitting.

$$L_{sup} = -\log \sum_{j=1}^{C} y_j f_j(x), \tag{1}$$

$$L_{neg} = \sum_{j=1}^{C} f_j(x) \log f_j(x), \tag{2}$$

$$L_{warm} = L_{sup} + L_{neg}. \tag{3}$$

Afterward, our method stops supervised learning and relies solely on consistency regularization.

Co-Pseudo-Labeling

Our method simultaneously trains two neural networks (denoted as $f(x; \theta_1)$ and $f(x; \theta_2)$), which collaboratively purify training labels for each other and obtain the pseudo-labels. The advantage of this approach is that it can effectively reduce the confirmation bias that can arise from mimicking the pre-trained model annotation comparing to the usage of self-generated pseudo-labels. In the following, we take the example of using $f(x; \theta_1)$ to provide pseudo-labels for $f(x; \theta_2)$.

Firstly, we attempt to divide the provided partial labels as valid or noisy, i.e., whether the ground-truth labels are in the candidate label sets. Drawing inspiration from the minimal-loss criterion (Arazo et al. 2019; Chen et al. 2019) which assumes that noise-free samples are easier to learn. We speculate that if the partial label of the current sample is valid, the model warm-up trained using supervised loss can predict the sample to a category within its candidate label set with a higher probability. Our method adopts two types of data augmentation (Sohn et al. 2020; Berthelot et al. 2020), i.e., weak data augmentation $\text{Aug}_w(\cdot)$ and strong data augmentation $\text{Aug}_s(\cdot)$. Details are shown in the Appendix.

Specifically, we calculate the following division loss over the predicted probabilities of all samples with weak data augmentation $\{L_{div}(\text{Aug}_w(x^i); \theta_1)\}_{i=1}^{N}$.

$$L_{div}(x; \theta) = -\log f_j(x; \theta), j = \underset{j \in \mathcal{Y}, y_j = 1}{\arg\max} f_j(x; \theta), \tag{4}$$

where $f_j(x; \theta)$ indicates the predicted probability on the j-th category of neural network with parameter θ, $\mathcal{Y}$ represents the label space and N is total number of training samples. Then, we use a two-component Gaussian mixture model (GMM)(Permuter, Francos, and Jermyn 2006) to fit the above losses to classify the whole training set into a partial split whose annotated partial labels are assumed to be valid with a probability w^i, and an unlabeled split whose annotated partial labels are assumed to be non-valid and discarded. We use $\mathcal{P} = \{(x^i, p^i)\}$ to denote the partial split, and $\mathcal{U} = \{x^i\}$ to denote the unlabeled split.

$p^i = (p_1^i, p_2^i, \ldots, p_C^i)$ is calculated by re-scaling the predicted label distribution of x^i with Eq.5, which eliminates the probabilities outside the candidate label set.

$$p_j^i = \frac{y_j^i \cdot f_j(\mathrm{Aug}_w(x^i); \theta_1)}{\sum_{k=1}^{C} y_k^i f_k(\mathrm{Aug}_w(x^i); \theta_1)}, \quad \text{for } j = 1, 2, \ldots, C. \tag{5}$$

For more robust and generalizable pseudo-labels, we combine the predicted label distributions from both networks to obtain the fused pseudo-labels p' for training. We combine p^i with the average predicted probabilities of K weakly-augmented inputs from $f(x; \theta_2)$. The confidences of the validity of the partial labels for the samples in the partial split, i.e. w^i, are taken as the fusion weights of the predicted label distributions from $f(x; \theta_1)$. For samples in the unlabeled split, we combine the average predicted probabilities from both networks and set both weights to 0.5. The fusion process can be expressed as:

$$p'^i = \begin{cases} w^i \cdot p^i + (1 - w^i) \cdot \bar{p}_2^i, & \text{if } x^i \in \mathcal{P}; \\ (\bar{p}_1^i + \bar{p}_2^i)/2, & \text{if } x^i \in \mathcal{U}. \end{cases} \tag{6}$$

Here, $\bar{p}_{1(2)}^i = \frac{1}{K} \sum_{k=1}^{K} f(\mathrm{Aug}_w(x^i); \theta_{1(2)})$ represents the average predicted probabilities of K weakly-augmented inputs of one of the two networks. Ablation experiments show that the exploitation of the unlabeled split is of crucial importance for achieving performance improvements.

Finally, the pseudo-labels are sharpened with a temperature of T to obtain more discriminative label distributions,

$$\tilde{p}_j^i = \frac{(p_j'^i)^{1/T}}{\sum_{k=1}^{C} (p_k'^i)^{1/T}}, \quad \text{for } j = 1, 2, \ldots, C. \tag{7}$$

Self-Training

We perform self-training between the obtained pseudo-labels and predicted label distribution as well as the pseudo-labels and prototypical similarity distribution. Firstly, we use the pseudo-labels as the training objective of the predicted class distribution. We choose the cross-entropy loss as the training objective on $\mathcal{P}$ and the mean square error on $\mathcal{U}$ due to its noisy-tolerant property:

$$L_{\mathrm{cr}}(x; \theta) = \begin{cases} -\sum_{j=1}^{C} \tilde{p}_j \log f_j(x; \theta), & \text{if } x \in \mathcal{P}; \\ \sum_{j=1}^{C} (\tilde{p}_j - f_j(x; \theta))^2, & \text{if } x \in \mathcal{U}. \end{cases} \tag{8}$$

Secondly, we adopt the prototypical similarity alignment which enforces the consistency between the pseudo-labels and the sample representation. We project the output representations of image x^i of the two neural networks to a shared embedding space through a two-layer MLP with L2 normalization, respectively (See Fig.1), obtaining $z^i = g(f(\mathrm{Aug}_w(x^i); \bar{\theta}))$, in which $g(\cdot)$ represents the MLP projector and $\bar{\theta}$ represents the neural network parameters θ, excluding the last fully-connected layer. During the training process, we maintain a cluster center for each category in the shared representation space that represents the representation of the category, called "prototype", denoted by $\{o_j\}_{j=1}^{C}$. We believe that different data augmentation variants of the same sample should maintain consistent distributions between label space and representation space. Specifically, our method calculate the similarity distribution over the representation of current image and class prototypes as $s^i = \mathrm{softmax}(z^i o_1, z^i o_2, \ldots, z^i o_C)$, which is then aligned to the pseudo-label distribution $\tilde{p}^i$. We choose KL-Divergence for samples in partial split $\mathcal{P}$, which have a much higher pseudo-accuracy and mean square error for samples in unlabeled split $\mathcal{U}$. The loss functions for prototypical similarity alignment can be written as:

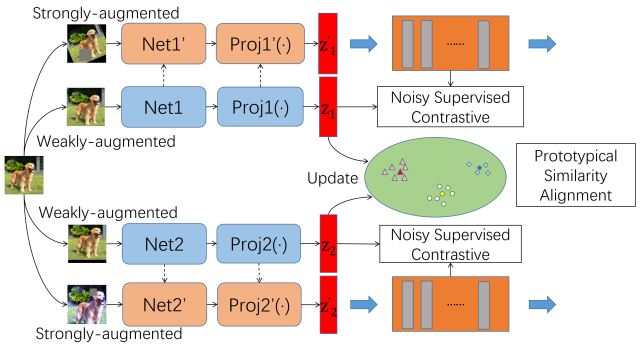

Figure 1: The illustration of prototypical similarity alignment and noisy supervised contrastive learning. The dotted lines represent momentum update.

$$L_{\mathrm{prot}}(x; \theta) = \begin{cases} \sum_{j=1}^{C} \tilde{p}_j^{T'} \log(\tilde{p}_j^{T'} / s_j^{T'}), & \text{if } x \in \mathcal{P}; \\ \sum_{j=1}^{C} (\tilde{p}_j - s_j)^2, & \text{if } x \in \mathcal{U}. \end{cases} \tag{9}$$

The class prototypes are momentum updated during training with Eq.10.

$$o_j = \gamma o_j + (1 - \gamma) z^i, \quad j = \arg\max_{j \in \mathcal{Y}} \tilde{p}_j \tag{10}$$

Noisy Contrastive Learning

To further exploit from the data distribution property of downstream unlabeled images while enhancing the model's representation ability, we employ the noisy supervised contrastive learning.

In addition, we utilize contrastive learning to pull together representations of samples from the same class while pushing apart those from different classes, enabling the model to encode more discriminative features on downstream data. In implementation, we adopt the MoCo (He et al. 2020) framework, in which a large-size "first-in-first-out" queue of image representations encoded by the momentum updated copy of our model is maintained. We select positive and negative set for the current image representation from the representation queue by their pseudo-labels and optimize the following noisy-tolerant contrastive loss:

$$\mathcal{L}_{\mathrm{ncont}}(x; \theta) = -\frac{1}{|P(x)|} \sum_{z_+ \in P(x)} \log \frac{\exp(z^\top z_+ / T'')}{\sum_{z_+ \in P(x)} \exp(z^\top z_+ / T'') + \sum_{z_- \in N(x)} \exp(z^\top z_- / T'')}, \tag{11}$$

Methods	CIFAR-10	CIFAR-100	SVHN	FMNIST	EuroSAT	FER2013	GTSRB
Zero-Shot(train)	88.40%	61.83%	9.34%	62.57%	32.26%	41.47%	24.87%
Zero-Shot*(train)	89.09%	62.75%	9.33%	65.81%	30.61%	46.55%	25.53%
Zero-Shot(test)	88.51%	61.55%	8.63%	61.46%	31.49%	42.07%	25.14%
Zero-Shot*(test)	89.01%	62.74%	8.82%	65.14%	30.78%	46.72%	25.57%
KD	87.39%	56.31%	8.01%	65.91%	35.24%	44.89%	25.70%
KD*	87.74%	56.80%	8.21%	67.60%	34.13%	46.43%	26.98%
DivideMix	93.32%	65.76%	16.46%	71.60%	42.41%	44.31%	30.07%
DivideMix*	93.83%	66.03%	17.16%	74.21%	37.74%	47.42%	32.69%
CR-DPLL	84.20%	60.05%	6.82%	71.27%	8.85%	16.75%	27.16%
ALIM-Onehot	93.18%	64.60%	17.06%	72.42%	34.57%	**52.77**%	31.06%
ALIM-Scale	93.59%	64.61%	20.66%	72.36%	37.11%	52.51%	31.81%
Co-Reg	**94.06**%	**71.04**%	**46.57**%	**76.28**%	**65.54**%	50.26%	**41.18**%
CoOp-2	74.25%	46.08%	15.10%	70.42%	53.74%	33.33%	22.53%
CoOp-4	75.19%	46.14%	20.90%	72.77%	60.44%	39.81%	20.01%
CoOp-8	75.66%	48.41%	28.18%	76.10%	68.52%	46.22%	21.02%
CoOp-16	75.00%	52.18%	27.14%	**78.99**%	**75.78**%	46.81%	25.55%

Table 1: Accuracy comparisons of all comparing methods on CLIP annotated datasets, best performances in bold. Zero-shot(train) and Zero-shot(test) separately indicate the performance of zero-shot learning in train and test dataset splits.

where $P(x)$ and $N(x)$ separately denote the set of selected positive and negative examples for image x, $T'' \geq 0$ is the temperature. We treat $x \in \mathcal{P}$ as confident samples and $x \in \mathcal{U}$ as lacking of confidence and applying the selection strategy for $P(x)$ and $N(x)$ in (Wang et al. 2024).

Experiments

Experimental Setup

We conduct experiments on several image classification benchmarks: CIFAR-10, CIFAR-100 (Krizhevsky, Hinton et al. 2009), SVHN (Goodfellow et al. 2013a), Fashion-MNIST (Xiao, Rasul, and Vollgraf 2017), EuroSAT (Helber et al. 2019), FER2013 (Goodfellow et al. 2013b) and GTSRB (Houben et al. 2013). We annotate the images of these datasets with CLIP ViT-B/32 in the manner of partial label or single label following Section 3. For CIFAR-10, CIFAR-100, Fashion-MNIST and EuroSAT, we use the class descriptions from PyTorch; and for SVHN, FER2013 and GTSRB, their class descriptions are manually assigned and are the same for all comparison methods. The prompt templates we used are listed in Appendix.

We compare the performances of our method with various types of WSL methods under partial and single label annotations: DivideMix (Li, Socher, and Hoi 2020), CR-DPLL (Wu, Wang, and Zhang 2022), ALIM-Onehot and ALIM-Scale (Xu et al. 2024), in which CR-DPLL, ALIM-Onehot and ALIM-Scale learn from partial label annotations and DivideMix learn from single label annotations. Also, we compare with other pre-trained application paradigms: zero-shot, unsupervised knowledge distillation (KD) and few-shot fine-tuning. Zero-shot, KD and WSL require no human labeling while few-shot fine-tuning is performed with 2, 4, 8, and 16 labeled samples per class. We choose prompt learning method CoOp (Zhou et al. 2022b) as the few-shot fine-tuning comparing method. For zero-shot learning, KD and DivideMix, we record the performances with single prompt template "a photo of a $\{\}$" as well as using the average of predicted probabilities of multiple prompt templates for fair comparison (superscript with asterisk).

The average amount of candidate labels per training sample and the proportions of ground-truth label being outside of the candidate sets, i.e. η, is recorded for partially annotated datasets in Table 2.

We use the PreAct ResNet-18 (He et al. 2016) as the backbone for all comparing methods. The training batch-size is 256, and the number of warm up and total epochs are chosen from 50 or 100 and 100 or 800, respectively. The number of weakly-augmented inputs for co-pseudo-labeling is $K = 2$ and the sharpening temperature is $T = 0.5$. The dimension of projected representations is 128, and the length of MoCo queue is 8192. The loss weights are $\lambda_1 = 0.1$, $\lambda_2 = 0.1$ The experiments are all carried on NVIDIA 3090 GPUs.

Main Results

Our method improves performance over zero-shot learning on all experimental datasets and outperforms CoOp on five of the seven datasets, even though CoOp uses human annotated task-relevant labels. On the other two datasets, CoOp performs worse than our method at 2 and 4 shots, while achieves better performances at 8 and 16 shots.

Among the comparing paradigms, only KD and WSL can obtain smaller-size deployment model, which is very flexible when facing application scenarios with restrained inference resources. However, student models distilled from the supervision of pre-trained teacher can hardly outperforms the teacher models without task-related training examples. Our method outperforms unsupervised KD on all experimental datasets.

Datasets	CIFAR-10	CIFAR-100	SVHN	FMNIST	EuroSAT	FER2013	GTSRB
Avg. Candi	1.39	2.36	2.41	1.58	3.26	2.38	2.84
η	4.79%	21.50%	61.61%	22.35%	32.89%	21.49%	58.22%

Table 2: The average number of candidate labels associated with each sample in partially annotated datasets, as well as the proportion of noisy partial labels.

Methods	$q = 0.01$			$q = 0.03$			$q = 0.05$		
	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$
CC	53.63%	48.84%	45.50%	51.85%	47.48%	43.37%	50.64%	45.87%	40.87%
RC	52.73%	48.59%	45.77%	52.15%	48.25%	43.92%	46.62%	45.46%	40.31%
LWC	53.16%	48.64%	45.51%	51.69%	47.60%	43.39%	50.55%	45.85%	39.83%
LWS	56.05%	50.66%	45.71%	53.59%	48.28%	42.20%	45.46%	39.63%	33.60%
PiCO	68.27%	62.24%	58.97%	67.38%	62.01%	58.64%	67.52%	61.52%	58.18%
CR-DPLL	68.12%	65.32%	62.94%	67.53%	64.29%	61.79%	67.17%	64.11%	61.03%
PiCO+	75.04%	74.31%	71.79%	74.68%	73.65%	69.97%	73.06%	71.37%	67.56%
IRNet	71.17%	70.10%	68.77%	71.01%	70.15%	68.18%	70.73%	69.33%	68.09%
ALIM-Scale	77.37%	76.81%	76.45%	77.60%	76.63%	**75.92**%	76.86%	**76.44**%	**75.67**%
ALIM-Onehot	76.52%	76.55%	76.09%	77.27%	76.29%	75.29%	**76.87**%	75.23%	74.49%
Co-Reg	**78.13**%	**78.01**%	**77.20**%	**77.16**%	**76.85**%	75.71%	76.30%	74.91%	73.45%

Table 3: Accuracy comparisons on synthetic NPLL datasets, best performances in bold.

w/o Co-PL	w/ SupCont	w/o proto	w/o U	Co-Reg
68.40%	67.96%	69.81%	65.32%	71.04%

Table 4: Ablation experiments on different degenerations of Co-Reg.

Our method outperforms other WSL methods on all datasets except one, on which ALIM-Onehot performs the best. The results demonstrate the effectiveness of our method and partial label annotation. CLIP's performance on the SVHN dataset is extremely poor. It predicts almost all samples as "number 0", making it impossible to obtain an effective model using ordinary training algorithms. However, though most prompt templates fail, one of them can achieve an accuracy rate over 25% and the correctly predicted labels are included into the candidate label sets, which facilitates the training of our method.

Synthetic Datasets

We also conduct the experiments on synthetic datasets of CIFAR-100, following the generation process used by the previous method (Xu et al. 2024). First, we generate partially labeled datasets by flipping negative labels $\bar{y} \neq y$ to false positive labels with a probability $q = P(\bar{y} \in Y | \bar{y} \neq y)$. Then, we generate noisy partially labeled datasets by randomly substituting the ground-truth label with a non-candidate label with a probability $\eta = P(y \notin Y)$ for each sample. We choose the noise level η from $\{0.1, 0.2, 0.3\}$, and consider $q \in \{0.01, 0.03, 0.05\}$ for CIFAR-100.

We compare our method with ten PLL and NPLL methods, i.e. CC (Feng et al. 2020), RC (Feng et al. 2020), LWC (Wen et al. 2021), LWS (Wen et al. 2021), PiCO (Wang et al. 2022a), CR-DPLL (Wu, Wang, and Zhang 2022), PiCO+ (Wang et al. 2022b), IRNet (Lian et al. 2022), ALIM-Scale and ALIM-Onehot (Xu et al. 2024).

On five of the nine subtasks, our method achieves the best performances, while on the remaining subtasks, ALIM-Onehot or ALIM-Scale achieves the best performances (See Table 3). It is worth noting that our method is not designed for synthetic datasets, but still achieves good performance. It can be clearly seen that our method has more advantages when q is small. This is because there are usually relatively few candidate labels associated with each sample on the dataset annotated by the pre-trained model.

Ablations

We conduct experiments on four degenerations of our method to demonstrate the effectiveness of our proposed modules, which are: 1. w/o Co-PL: replaces the collaborative pseudo-labeling mechanism to performing pseudo-labeling with their own prediction; 2. w/ SupCont: replace noisy supervised contrastive learning to traditional supervised contrastive learning; 3. w/o proto: does not perform prototypical similarity alignment; 4. w/o U: discarding unlabeled set $\mathcal{U}$ during training. It can be seen that, all modules contribute positively to the performance of our method.

Discussion and Limitation

In this section, we briefly discuss incorporating weakly supervised learning into distillation from pre-trained models and compare this approach with other mainstream paradigms of applying pre-trained models to downstream tasks.

Paradigms	Samples	Human Annotations	Inference Size	Perf. Improvements
Zero-Shot	×	×	-	-
Prompt Learning / Adapter	few	few	increase sightly	✓
KD_{unsup}	✓	×	small	×
KD_{sup}	✓	✓	small	✓
Fully Fine-Tuning	✓	✓	-	✓
WSL	✓	×	small	✓

Table 5: Comparison among different pre-trained model application paradigms. Zero-Shot means directly performing zero-shot inference on untrained tasks. Prompt Learning fine-tunes a parameterized prompt, which is concatenated with text or image inputs, using few-shot downstream samples and Adapter adds a small number of trainable parameters to the neural network. KD_{sup} and KD_{unsup} represent supervised and unsupervised knowledge distillation, i.e. with or without task labels, respectively. "-" means remaining the same with original model.

Advantages of Incorporating Weakly Supervised Learning

Just like what this paper does, we can use pre-trained models as weak annotators to annotate unlabeled samples of downstream task with weak labels, and then formalize this task as a specific type of weakly supervised learning and design corresponding algorithms to address it.

Table 5 compares different pre-trained model application paradigms. It is evident that WSL is the only one that can achieve performance improvements over the original model without using additional manual annotations. Meanwhile, by retraining specialized small models on the downstream samples, the inference model size are significantly reduced. Additionally, due to the fact that few-shot fine-tuning techniques (e.g., prompt learning and adaptors) only add a small number of trainable parameters to the pre-trained models, their performance improvements are usually limited.

It is worth noting that the main difference from traditional unsupervised knowledge distillation is that this approach formalizes the downstream task as a specific weakly supervised learning problem and employs corresponding algorithms and can achieve significantly better performances on many scenarios compared to the pre-trained model. In contrast, knowledge distillation uses the output class distributions of the pre-trained model as training target, aiming to transfer the knowledge from a large model to a smaller specialized model and often does not achieve performance improvements over the pre-trained model.

Limitation

Nevertheless, there are two main limitations associated with using weakly supervised learning. First, in downstream tasks where the training images are relatively similar to the general domain images used for pre-training the large model, such as ImageNet, specialized models trained through weakly supervised learning often fail to surpass the performance of the original pre-trained model. In these tasks, the large model is already highly effective, and zero-shot inference typically yields satisfactory results. Our method should primarily be applied to tasks where the image domain significantly differs from the general domain, and where the pre-trained model does not perform well.

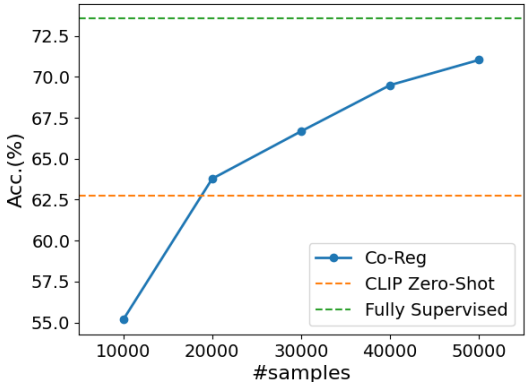

Figure 2: Accuracy on CIFAR-100 using different numbers of unlabeled samples.

Second, applying weakly supervised learning requires a large number of downstream unlabeled samples, which we assume are readily available. Here we conduct experiments on CIFAR-100 using different numbers of unlabeled samples and observe the performance of our method to explore its dependence on the number of unlabeled samples. As shown in Fig.2, our method requires 20,000 unlabeled samples (i.e., 200 per class) on CIFAR-100 to exceed the performance of CLIP zero-shot; when using the complete dataset, our method only drops 3% in performance compared to fully supervised training.

Conclusion

Against the backdrop of significant advances in pre-trained large models, this paper makes a modest attempt to explore whether WSL can shift from primarily focusing on learning from manually annotated weak labels to using weak labels provided by pre-trained models and also examines whether this approach offers advantages compared to other pre-trained model application paradigms. In some image classification tasks, we use CLIP to annotate partial labels for images and designed NPLL algorithms for training, achieving effective results.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China under grant 62171248, the PCNL KEY project (PCL2023A08) and E Fund Management Co., Ltd.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Arazo, E.; Ortego, D.; Albert, P.; O'Connor, N.; and McGuinness, K. 2019. Unsupervised label noise modeling and loss correction. In *International conference on machine learning*, 312–321. PMLR.

Bao, H.; Wang, W.; Dong, L.; Liu, Q.; Mohammed, O. K.; Aggarwal, K.; Som, S.; Piao, S.; and Wei, F. 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems*, 35: 32897–32912.

Berthelot, D.; Carlini, N.; Cubuk, E. D.; Kurakin, A.; Sohn, K.; Zhang, H.; and Raffel, C. 2020. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *International Conference on Learning Representations*.

Chen, P.; Liao, B. B.; Chen, G.; and Zhang, S. 2019. Understanding and utilizing deep neural networks trained with noisy labels. In *International conference on machine learning*, 1062–1070. PMLR.

Feng, L.; and An, B. 2018. Leveraging Latent Label Distributions for Partial Label Learning. In *IJCAI*, 2107–2113.

Feng, L.; Lv, J.; Han, B.; Xu, M.; Niu, G.; Geng, X.; An, B.; and Sugiyama, M. 2020. Provably Consistent Partial-Label Learning. In *Advances in neural information processing systems*.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2024. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595.

Goodfellow, I. J.; Bulatov, Y.; Ibarz, J.; Arnoud, S.; and Shet, V. 2013a. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *arXiv preprint arXiv:1312.6082*.

Goodfellow, I. J.; Erhan, D.; Carrier, P. L.; Courville, A.; Mirza, M.; Hamner, B.; Cukierski, W.; Tang, Y.; Thaler, D.; Lee, D.-H.; et al. 2013b. Challenges in representation learning: A report on three machine learning contests. In *Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III 20*, 117–124. Springer.

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

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, 630–645. Springer.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7): 2217–2226.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Houben, S.; Stallkamp, J.; Salmen, J.; Schlipsing, M.; and Igel, C. 2013. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In *The 2013 international joint conference on neural networks (IJCNN)*, 1–8. Ieee.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19113–19122.

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

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, J.; Socher, R.; and Hoi, S. C. 2020. DivideMix: Learning with Noisy Labels as Semi-supervised Learning. In *International Conference on Learning Representations*.

Lian, Z.; Xu, M.; Chen, L.; Sun, L.; Liu, B.; and Tao, J. 2022. Arnet: Automatic refinement network for noisy partial label learning. *arXiv preprint arXiv:2211.04774*.

Lv, J.; Xu, M.; Feng, L.; Niu, G.; Geng, X.; and Sugiyama, M. 2020. Progressive Identification of True Labels for Partial-Label Learning. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, 6500–6510. PMLR.

Lyu, G.; Feng, S.; Wang, T.; Lang, C.; and Li, Y. 2021. GM-PLL: Graph Matching Based Partial Label Learning. *IEEE Trans. Knowl. Data Eng.*, 33(2): 521–535.

Permuter, H.; Francos, J.; and Jermyn, I. 2006. A study of Gaussian mixture models of color and texture features for image classification and segmentation. *Pattern recognition*, 39(4): 695–706.

Qiao, C.; Xu, N.; Lv, J.; Ren, Y.; and Geng, X. 2023. Fredis: A fusion framework of refinement and disambiguation for unreliable partial label learning. In *International Conference on Machine Learning*, 28321–28336. PMLR.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. *arXiv preprint arXiv:2103.00020*.

Shi, Y.; Wu, D.-D.; Geng, X.; and Zhang, M.-L. 2023a. Robust Representation Learning for Unreliable Partial Label Learning. *arXiv preprint arXiv:2308.16718*.

Shi, Y.; Xu, N.; Yuan, H.; and Geng, X. 2023b. Unreliable partial label learning with recursive separation. *arXiv preprint arXiv:2302.09891*.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C.; Cubuk, E. D.; Kurakin, A.; and Li, C. 2020. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *NeurIPS*.

Wang, H.; Xiao, R.; Li, Y.; Feng, L.; Niu, G.; Chen, G.; and Zhao, J. 2022a. PiCO: Contrastive Label Disambiguation for Partial Label Learning. *International Conference on Learning Representations*.

Wang, H.; Xiao, R.; Li, Y.; Feng, L.; Niu, G.; Chen, G.; and Zhao, J. 2022b. PiCO+: Contrastive Label Disambiguation for Robust Partial Label Learning. *arXiv preprint arXiv:2201.08984*.

Wang, H.; Yang, S.; Lyu, G.; Liu, W.; Hu, T.; Chen, K.; Feng, S.; and Chen, G. 2023. Deep Partial Multi-Label Learning with Graph Disambiguation. *arXiv preprint arXiv:2305.05882*.

Wang, Q.-W.; Li, Y.-F.; and Zhou, Z.-H. 2019. Partial Label Learning with Unlabeled Data. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 3755–3761.

Wang, Q.-W.; Zhao, B.; Zhu, M.; Li, T.; Liu, Z.; and Xia, S.-T. 2024. Controller-Guided Partial Label Consistency Regularization with Unlabeled Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15571–15579.

Wen, H.; Cui, J.; Hang, H.; Liu, J.; Wang, Y.; and Lin, Z. 2021. Leveraged Weighted Loss for Partial Label Learning. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, 11091–11100. PMLR.

Wu, D.-D.; Wang, D.-B.; and Zhang, M.-L. 2022. Revisiting Consistency Regularization for Deep Partial Label Learning. In *International conference on machine learning*.

Xia, S.; Lv, J.; Xu, N.; Niu, G.; and Geng, X. 2023. Towards Effective Visual Representations for Partial-Label Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15589–15598.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Xu, M.; Lian, Z.; Feng, L.; Liu, B.; and Tao, J. 2024. ALIM: Adjusting Label Importance Mechanism for Noisy Partial Label Learning. *Advances in Neural Information Processing Systems*, 36.

Xu, N.; Qiao, C.; Geng, X.; and Zhang, M.-L. 2021. Instance-dependent partial label learning. *Advances in Neural Information Processing Systems*, 34: 27119–27130.

Zhang, M.; and Yu, F. 2015. Solving the Partial Label Learning Problem: An Instance-Based Approach. In *IJCAI*, 4048–4054. AAAI Press.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.