Targeted Representation Alignment for Open-World Semi-Supervised Learning (Supplementary Material)

Ruixuan Xiao[1], Lei Feng[2], Kai Tang[1], Junbo Zhao[1], Yixuan Li[3], Gang Chen[1], Haobo Wang[1]*
[1]Zhejiang University, China
[2]Singapore University of Technology and Design, Singapore
[3]University of Wisconsin-Madison, USA
{xiaoruixuan,tk0819,j.zhao,cg,wanghaobo}@zju.edu.cn, lfengqaq@gmail.com, sharonli@cs.wisc.edu

Appendix

In this appendix, we further provide the following contents:

- Additional details of rough assignment in A.1.
- Optimizations of the auxiliary head in A.2.
- Additional details of contrastive learning in A.3.
- Additional implementation details in A.4.
- Additional feature visualizations in B.1.
- Sensitivity analysis of parameters, including the estimated class number K and temperature τ in B.2.
- Additional results with less labeled data in B.3.
- Results on generic datasets with ViT backbone in B.4.
- Effect of different PU learning methods in B.5.
- Effect of ETF structure initialization in B.6.
- Additional visualiztions of NC in B.7.
- Further enhancement with SSL methods in B.8.
- Analysis of training complexity in B.9.
- Details of benchmark datasets in B.10.
- Literature of Positive-Unlabeled learning in C.1.
- Comparisons with related settings in C.2.
- Overall algorithm of training pipeline in D.

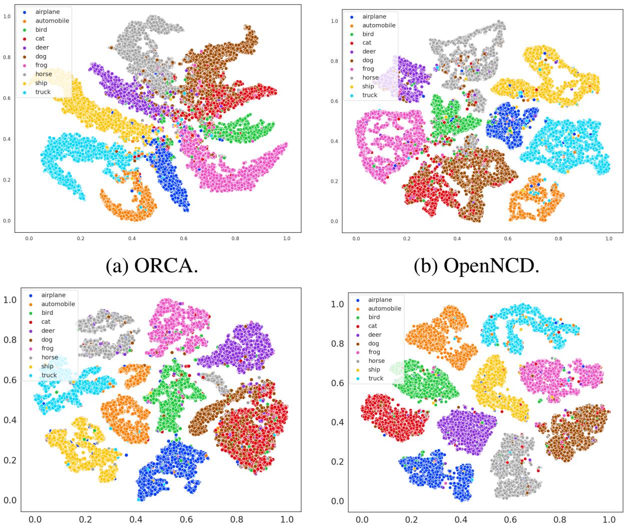
A. More Details of Implementation

In this section, we describe more details of implementations for our proposed TRAILER as follows.

A.1. Additional Details of Rough Assignment

As discussed in Section 3.2 of the main text, we adopt an equipartition constraint [5, 17] to induce a rough label assignment matrix $\mathbf{Q}$, which is achieved by solving an optimal transport problem formulated by,

$$\mathbf{Q} = \max_{\mathbf{Q}\in\Gamma} \operatorname{Tr}(\mathbf{Q}^\top \mathbf{P}) + \epsilon \operatorname{H}(\mathbf{Q})$$
$$\text{s.t. } \Gamma = \{\mathbf{Q} \in \mathbb{R}_+^{K\times b} \mid \mathbf{Q}\mathbf{1}_b = \frac{1}{K}\mathbf{1}_k, \mathbf{Q}^\top \mathbf{1}_K = \frac{1}{b}\mathbf{1}_b\} \tag{11}$$



(a) ORCA.

(b) OpenNCD.

(c) TRAILER without calibration.

(d) TRAILER with calibration.

Figure 8. T-SNE feature visualization on CIFAR-10 for more baselines. Different colors represent the corresponding ground-truth classes. ORCA and OpenNCD also suffer from representation collapse, where the dog and cat classes are inseparable.

In Eq. (11), we aim to search for assignment $\mathbf{Q}$ close to logit $\mathbf{P}$ by maximizing $\operatorname{Tr}(\mathbf{Q}^\top\mathbf{P})$, while subject to Γ. In constraint Γ, we adopt an equipartition item $\mathbf{Q}\mathbf{1}_b = \frac{1}{K}\mathbf{1}_K$, which enforces each class is selected $\frac{b}{K}$ times uniformly in the batch. This avoids degenerate solutions of falling into the same class. Notably, the entropy regularizer $\operatorname{H}(\cdot)$ is included to make the resulting objective smoothing and convex, thus reducing the computation cost. This problem can be solved by the well-known Sinkhorn-Knopp algorithm [11] for efficient optimization (please refer to [1] for the complete details). Formally, we define a matrix $\mathbf{M} = \exp(\frac{\mathbf{P}}{\epsilon})$ which is the element-wise exponential of

*Corresponding author.

Methods	CIFAR-10			CIFAR-100			ImageNet-100		
	Known	Novel	All	Known	Novel	All	Known	Novel	All
FixMatch [34]	64.3	49.4	47.3	30.9	18.5	15.3	60.9	33.7	30.2
DS^{3}L [20]	70.5	46.6	43.5	33.7	15.8	15.1	64.3	28.1	25.9
DTC [21]	42.7	31.8	32.4	22.1	10.5	13.7	24.5	17.8	19.3
RankStats [22]	71.4	63.9	66.7	20.4	16.7	17.8	41.2	26.8	37.4
SimCLR [10]	44.9	48.0	47.7	26.0	28.8	26.5	42.9	41.6	41.5
ORCA [4]	82.8	85.5	84.1	52.5	31.8	38.6	83.9	60.5	69.7
GCD [37]	78.4	79.7	79.1	49.7	27.6	38.0	82.3	58.3	68.2
OpenLDN [33]	73.1	90.1	82.1	37.0	33.4	35.1	51.5	37.0	43.9
OpenNCD [30]	83.5	86.7	85.3	53.6	33.0	41.2	84.0	65.8	73.2
TRAILER (Ours)	**87.6**	**91.3**	**89.6**	**53.9**	**42.9**	**48.2**	**87.1**	**76.5**	**81.5**

Table 5. Accuracy comparison of known, novel, and all classes on CIFAR-10, CIFAR-100 and ImageNet-100 dataset. The dataset is composed of 50% known classes and 50% novel classes, with **only 10%** of the known classes labeled.

$\mathbf{P}/\epsilon$. The label assignment is obtained by,

$$\mathbf{Q}^* = \operatorname{diag}(\boldsymbol{u})\mathbf{M}\operatorname{diag}(\boldsymbol{v}) \tag{12}$$

Where $\boldsymbol{u} \in \mathbb{R}^K$ and $\boldsymbol{v} \in \mathbb{R}^b$ are renormalization vectors to make $\mathbf{Q}^*$ a probability matrix and are updated iteratively by,

$$\forall y : \boldsymbol{u} \leftarrow [\mathbf{M}\boldsymbol{v}]_y^{-1}, \quad \forall i : \boldsymbol{v} \leftarrow [\boldsymbol{u}^\top \mathbf{M}]_i^{-1} \tag{13}$$

where $1 \leq y \leq k$ and $1 \leq i \leq b$. This is known as the Sinkhorn's fixed point iteration problem. In practice, we adopt a small iteration number of 3 following [17].

A.2. Optimizations of Auxiliary Head h_{aux}

Recall in the label refinery procedure, we adopt an auxiliary head h_{aux} to address the binary task of known-novel separation and cast this task to the Positive-Unlabeled learning paradigm. Here we provide the details of the variational optimization algorithm [9] we adopted.

Formally, for this simplified task of binary classification between the positive (known) and negative (novel) classes. The output space is denoted as $Y \in \{+1, 0\}$ where $+1$ and 0 indicate the sample is positive (known) and negative (novel) respectively. The marginal distributions of the positive, negative, and unlabeled classes are formulated as $P_p(x) = P(x \mid y = +1)$, $P_n(x) = P(x \mid y = 0)$, and $P(x)$ respectively. The training data comprise a labeled positive dataset $\mathcal{P} = \{(x_i, y_i = +1)\}_{i=1}^{N_p} \overset{\text{i.i.d}}{\sim} P_p(x)$ containing merely positive samples and an unlabeled dataset $\mathcal{U} = \{x_i\}_{i=1}^{N_u} \overset{\text{i.i.d}}{\sim} P(x)$ containing both positive and negative. We aim to learn a binary auxiliary classifier $h_{aux}(x)$ that is parametric to approximate the ideal Bayesian classifier $h^*_{aux}(x) \triangleq P(y = +1 \mid x)$, from $\mathcal{P}$ and $\mathcal{U}$. The positive distribution $\hat{P}_p(x)$ can be estimated using the Bayes rule:

$$\begin{aligned} P_p(x) &= \frac{P(y=+1 \mid x)P(x)}{\int P(y=+1 \mid x)P(x)dx} \\ &\approx \frac{h_{aux}(x)P(x)}{E_u[h_{aux}(x)]} \triangleq \hat{P}_p(x) \end{aligned} \tag{14}$$

We can further prove that $P_p(x) = \hat{P}_p(x)$ if and only if $h_{aux}(x) = h^*_{aux}(x)$, under the Assumption 1. Please refer to [9] for the detailed proof.

Assumption 1 *There exists a set* $\mathcal{A} \subset \mathbb{R}^d$ *satisfying* $\int_{\mathcal{A}} P_p(x)dx > 0$ *and*

$$h^*_{aux}(x) = 1, \forall x \in \mathcal{A} \tag{15}$$

After that, the quality of $\hat{P}_p(x)$ can be evaluated by the Kullback-Leibler (KL) divergence as follows,

$$\begin{aligned} & \mathrm{KL}(P_p(x) \| \hat{P}_p(x)) \\ = \; & E_p[\log(P_p(x)) - \log(\hat{P}_p(x))] \\ = \; & E_p\left[\log\left(h^*_{aux}(x)\right)\right] - \log\left(E_u\left[h^*_{aux}(x)\right]\right) \\ & - E_p[\log(h_{aux}(x))] + \log\left(E_u[h_{aux}(x)]\right) \\ = \; & \mathcal{L}_{var}(h_{aux}(x)) - \mathcal{L}_{var}(h^*_{aux}(x)) \end{aligned} \tag{16}$$

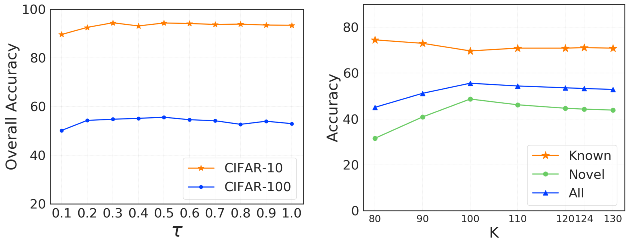
The derivation of Eq. (16) is based on the definition of KL divergence and Eq. (14). $\mathcal{L}_{var}$ is the variational loss formulated as,

$$\mathcal{L}_{var}(h_{aux}(x)) = \log\left(E_u[h_{aux}(x)]\right) - E_p[\log(h_{aux}(x))] \tag{17}$$

Stemming from the non-negative property of KL divergence, $\mathcal{L}_{var}(h_{aux}(x))$ is the variational upper bound of $\mathcal{L}_{var}(h^*_{aux}(x))$. Hence the minimization of Eq. (16) can be accomplished by minimizing $\mathcal{L}_{var}(h_{aux}(x))$, which can

Methods	CIFAR-10			CIFAR-100			ImageNet-100		
	Known	Novel	All	Known	Novel	All	Known	Novel	All
k-means	85.7	82.5	83.6	52.2	50.8	52.0	75.5	71.3	72.7
RankStat+ [22]	19.2	60.5	46.8	77.6	19.3	58.2	61.6	24.8	37.1
UNO+ [17]	**98.3**	53.8	68.6	80.6	47.2	69.5	**95.0**	57.9	70.3
ORCA [4]	86.2	79.6	81.8	77.4	52.0	69.0	92.6	63.9	73.5
GCD [37]	97.9	88.2	91.5	76.2	66.5	73.0	89.8	66.3	74.1
GPC [43]	98.2	89.1	92.2	**85.0**	63.0	77.9	94.3	71.0	76.9
TRAILER (Ours)	95.7	**98.3**	**97.5**	83.5	**74.3**	**80.5**	94.7	**82.1**	**86.3**

Table 6. Results with **ViT backbone** following [37] on CIFAR-10, CIFAR-100 and ImageNet-100. 50% of known samples are labeled.



(a) Ablation of different values of τ. (b) Ablation of K on CIFAR-100.

Figure 9. (a) Overall accuracy with different τ from 0.1 to 1.0 on CIFAR-10 and CIFAR-100. (b) Ablation results with different estimated class numbers K from 80 to 130 on CIFAR-100.

be empirically calculated with the averages over the positive and unlabeled samples in a prior-free manner,

$$\hat{\mathcal{L}}_{var} = \log \frac{\sum_{i=1}^{n_u} h_{aux}\left(x_i^u\right)}{n_u} - \frac{\sum_{i=1}^{n_p} \log\left(h_{aux}\left(x_i^p\right)\right)}{n_p} \tag{18}$$

where n_p and n_u are the numbers of positive and negative samples respectively in the mini-batch. A consistency regularization $\mathcal{L}_{reg}$ is also adopted to improve the robustness through sample mixup,

$$\begin{aligned} \tilde{x} &= \gamma \cdot x^p + (1-\gamma) \cdot x^u \\ \tilde{h} &= \gamma \cdot 1 + (1-\gamma) \cdot h_{aux}\left(x^u\right) \\ \mathcal{L}_{reg} &= E[(\log(\tilde{h}) - \log h_{aux}(\tilde{x}))^2] \end{aligned} \tag{19}$$

where $\gamma \sim \text{Beta}(\varsigma, \varsigma)$ and ς is a hyperparameter. The integral loss $\mathcal{L}_{aux}$ for optimizing h_{aux} is then formulated,

$$\mathcal{L}_{aux} = \hat{\mathcal{L}}_{var} + \lambda_{reg}\mathcal{L}_{reg} \tag{20}$$

The hyperparameters λ_{reg} and ς are set following [9]. Notably, in our practical implementation, the auxiliary head $h_{aux}(x)$ shares the same representation from the backbone and does not propagate gradients back to it.

PU algorithm	CIFAR-10			CIFAR-100		
	Known	Novel	All	Known	Novel	All
None (w/o h_{aux})	**95.2**	89.6	91.5	70.3	45.4	53.7
Dist-PU [45]	94.6	92.3	93.1	70.6	45.9	54.3
nnPU [26]	94.4	94.1	94.2	**71.1**	48.1	**55.7**
VPU (ours) [9]	93.4	**95.0**	**94.4**	69.7	**48.7**	55.6

Table 7. Results of TRAILER when equipped with different PU learning algorithms for known-novel separation in label refinery.

A.3. Additional Details of Contrastive Learning

During unsupervised contrastive learning, in addition to the self-contrastive objective $\mathcal{L}_{con}^{self} = -\log \frac{\exp(z_i^\top z_i'/\tau)}{\sum_{j=1}^n \exp(z_i^\top z_j'/\tau)}$, we also adopt a nearest neighbor sampling strategy following [16] to increase the diversity of the positive support set. Specifically, we collect k-nearest neighbors within embedding space for each sample feature z_i in the batch as $\mathcal{N}_k(z_i)$, then the contrastive objective based on nearest neighbors instead of the augmented view is formulated,

$$\mathcal{L}_{con}^{nn} = \sum_{z_i^{nn} \in \mathcal{N}_k(z_i)} -\log \frac{\exp(z_i^\top z_i^{nn}/\tau)}{\sum_{j=1}^n \exp(z_i^\top z_j'/\tau)} \tag{21}$$

Notably, this calculation does not require extra storage as embeddings of all samples have been stored for prototype update. The total contrastive loss is then aggregated,

$$\mathcal{L}_{con} = \mathcal{L}_{con}^{self} + \lambda_{nn}\mathcal{L}_{con}^{nn} \tag{22}$$

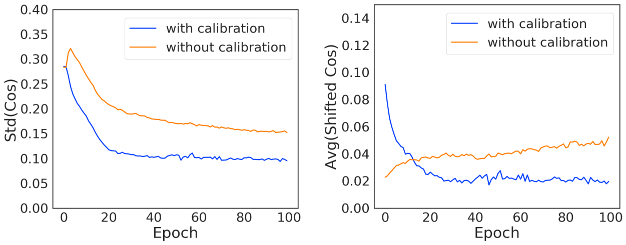
where λ_{nn} is a parameter ramped up from 0 to 1 in training.

A.4. Additional Implementation Details

Here we provide additional implementation details in our experiments. For CIFAR experiments, the loss weight α is set as 5 in the warm-up phase and 1 in the subsequent training process. The filtering rate $R\%$ is ramped up from 0.3 to 0.9 for CIFAR-10 and 0.5 to 1 for CIFAR-100. The calculation of unlabeled classification loss and prototype update is

Methods	CIFAR-10			CIFAR-100		
	Known	Novel	All	Known	Novel	All
TRAILER	**93.4**	95.0	94.4	69.7	**48.7**	**55.6**
w/ Reallocate	93.3	92.3	92.6	**74.5**	45.0	54.9
w/ Rotation	93.0	**95.9**	**94.9**	69.6	46.3	54.1

Table 8. Results with different ETF initialization strategies on CIFAR-10 and CIFAR-100.



(a) Equiangularity

(b) Maximal separation

Figure 10. Visualizations on the (a) equiangularity and (b) maximal separation properties of neural collapse on CIFAR-100.

then only performed on the selected set $\mathcal{D}_{sel}$. The pseudo-label filtering is performed at the start of each epoch, instead of batch-wise manner. The number of neighbors k for neighbor augmented contrastive learning is set as 50/10 for CIFAR-10/CIFAR-100. For ImageNet-100, we use ResNet-50 as the backbone with a batch size of 512 and a learning rate of $1e^{-2}$ for training of 120 epochs. We adopt standard SGD as the optimizer with the momentum of and weight decay of $1e^{-4}$. For the fine-grained benchmarks, we follow GCD [37] and adopt a ViT-B/16 backbone [13] pre-trained with DINO [6]. We use the output of `[CLS]` token with 768 dimensions as the embedding and train with a batch size of 128 for 200 epochs. The initial learning rate is set as 0.1 and α is fixed as 2. The vanilla softmax function is adopted for rough assignment on Herbarium 19. The optimization parameters for the auxiliary head and the contrastive learning procedure follow previous work [9] and [36] respectively. For experiments on ImageNet and semantic shift benchmarks, we choose OpenCon [36] and SimGCD [38] as our base implementation and the vanilla softmax predictions are adopted during the coarse assignment. Our core algorithm is developed using PyTorch [32] and we conduct all the experiments with NVIDIA RTX A6000 GPUs.

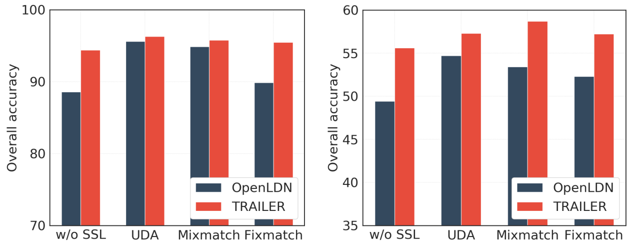
B. Additional Experimental Results

B.1. Additional Feature Visualizations for Baselines

We further provide the representation visualizations with t-SNE on the CIFAR-10 dataset for baselines ORCA and OpenNCD in Figure 8. It can be observed that ORCA and OpenNCD also suffer from the representation collapse dilemma, where the samples from cat (red, known), dog classes (brown, novel), and even part of horse classes (grey, novel) are intertwined and indistinguishable.

Methods	CIFAR-10			CIFAR-100		
	Known	Novel	All	Known	Novel	All
TRAILER	93.4	95.0	94.4	69.7	48.7	55.6
w/ Fixmatch	95.0	95.7	95.5	70.6	50.6	57.2
w/ Mixmatch	95.3	96.1	95.8	74.1	**51.1**	**58.7**
w/ UDA	**95.9**	**96.5**	**96.3**	**74.2**	49.2	57.3

Table 9. Results when equipped with different SSL methods.



(a) CIFAR-10

(b) CIFAR-100

Figure 11. Comparisons of overall accuracy equipped with different SSL methods on CIFAR-10/CIFAR-100. 'w/o SSL' indicate the vanilla performances without second stage of SSL training.

B.2. Sensitivity Analysis for Hyperparameters.

Temperature Parameter τ. We investigate the effect of the temperature parameter τ on CIFAR-10 and CIFAR-100 datasets. Figure 9a shows the performance of TRAILER with different τ ranging from 0.1 to 1.0 for 'All' classes. It can be observed that TRAILER works well and attains stable performances in a wide range of τ for both CIFAR-10 and CIFAR-100. So we set a moderate value of τ without intensive tuning in our practical implementation.

Estimated Class Number K. Recall that we have verified the robustness of TRAILER with the estimated class number $K = 124$ on CIFAR-100 in the main text. Here we further provide more results with estimation error from -20% (underestimated, $K = 80$) to 30% (overestimated, $K = 130$). As shown in Figure 9b, when K is overestimated, the performance of TRAILER is stable and satisfactory over a wide range of K. Further, when K is underestimated, TRAILER bears a perceptible performance drop for 'Novel' classes, but achieves slightly better performance for 'Known' classes and remains competitive for 'All' classes. Interestingly, an overestimated K induces less performance drop than an underestimated one, as the model can adaptively activate a smaller number of proto-

Param ϵ	0.02	0.03	0.04	**0.05**	0.06	0.07	0.08	0.09
CIFAR-10	94.6	94.5	94.7	**94.4**	94.1	93.9	94.8	95.0

Table 10. Ablation results of the entropy weight ϵ for sinkhorn.

types when K is overestimated (only activate 112 when K=124), and such phenomenon has also been observed in previous works [4, 33]. Overall, such results indicate that TRAILER is able to achieve satisfactory performance when given a rough class number estimation using different potential off-the-shelf alternatives in real-world deployment.

Entropy weight ϵ in Eq. (7). As discussed in Section 4.1, we simply set the entropy weight $\epsilon = 0.05$ following previous work. We further provide its ablations in Table 10 below to show the stable performance of TRAILER.

B.3. Results with Reduced Labeled Data.

Instead of utilizing 50% of known class data for the labeled set, we further evaluate the performance of TRAILER with only 10% of known class samples as labeled set. As shown in Table 5, TRAILER is able to retain substantial improvements over all the counterparts with less labeled data. Especially for novel classes, while other baselines mostly exhibit obvious performance drops for novel classes, TRAILER consistently remains competitive and shows robustness.

B.4. Results with ViT for Generic Datasets.

We have provided the results with ViT backbone on fine-grained datasets in Section 4.3 of the main text, here we further display the comparisons on generic datasets using ViT backbone with DINO pre-trained weights following the protocol in [37]. Notably, the data split is different for CIFAR-100 as discussed later in Appendix B.10. As shown in Table 6, TRAILER improves upon the best baseline by 5.3%, 2.6%, and 9.4% overall accuracy on three datasets, which continuously verifies its effectiveness with transformer-based backbones for generic datasets.

B.5. Effect of Different PU Learning Methods.

For the label refinery component, we cast the known-novel separation task to the PU learning paradigm and adopt a prior-free variational algorithm VPU [9] to address it. Here we further investigate some other PU learning approaches, including Dist-PU [45] and nnPU [26]. As shown in Table 7, TRAILER is able to largely advance the performance equiped with different PU algorithms for label refinery. Overall, the label refinery component is designed as a flexible mechanism and we adopt an empirically-strong and easy-to-use VPU algorithm for this task of known-novel separation. One may design more powerful PU algorithms for this binary task, which we leave for future work.

Ablation	ORCA	OpenNCD	**TRAILER**	w/o h_{aux}	w/o OT
Epoch Time	50.50s	38.29s	43.15s	40.62s	42.73s

Table 11. Comparisons of the epoch-wise training time (seconds).

B.6. Effect of ETF Structure Initialization.

We also explore different initialization manners of the targeted classifier h_{etf}. In vanilla TRAILER, we simply preassign h_{etf} with a random simplex ETF without sophisticated initializations. Here we equip TRAILER with different initialization strategies: 1) *TRAILER with Reallocate* which adopt Hungarian match between embedding centers μ and the ETF vectors ϕ^{etf} to reallocate each targeted vector to its matching embedding center, at the first epoch after a warm-up period with learnable classifiers and retain this reallocation subsequently; 2) *TRAILER with Rotation* which follows recent work [18] and introduces a learnable orthogonal matrix to adjust directions of ETF structure (Simplex EFT becomes Trivial ETF, remains equiangular but losses the maximally separated property).

As shown in Table 8, TRAILER with Reallocate slightly underperforms the vanilla TRAILER for overall performance. Interestingly, on CIFAR-100 we observe that this variant obtains stronger performance on known classes, but at the expense of much weaker performance on novel classes. We speculate that the representations already get overfitted towards known classes in the warm-up phase with learnable classifier. On the other hand, TRAILER with Rotation excels on CIFAR-10 but lags behind on CIFAR-100, which indicates direction adjustment is beneficial for CIFAR-10 while maximal separation might be more important for larger label space. Overall, the random initialization that we adopt yields satisfactory and well-balanced results, and it can produce stable performance without fluctuation. We speculate such stability arises from progressive alignment and accurate target assignment.

B.7. Additional Visualizations of NC.

We further provide the visualizations of neural collapse on CIFAR-100 dataset, including $\mathrm{Std}_{k\neq k'}(\cos(\hat{z}^k, \hat{z}^{k'}))$ for equiangularity and $\mathrm{Avg}_{k\neq k'}(\cos(\hat{z}^k, \hat{z}^{k'}) + \frac{1}{K-1})$ for maximal separation, as discussed in Section 4.3 of main text. As shown in Figure 10, the standard deviations of cosines and shifted average cosine values exhibit a similar trend as on CIFAR-10 and approach near 0, which consistently indicates that TRAILER achieves the uniformly and maximally separated feature structure close to neural collapse. To prove the fitting degree between the feature center and ETF classifier for each class, we further calculate another metric $\mathrm{Avg}_{1\leq k\leq K}(\cos(\hat{z}^k, \phi_k^{etf}))$, which initially starts at only 0.57 and finally increases to over 0.99.

Dataset	Labeled $\mathcal{D}_l$ #Class	Labeled $\mathcal{D}_l$ #Image	Unlabeled $\mathcal{D}_u$ #Class	Unlabeled $\mathcal{D}_u$ #Image
CIFAR-10	5	12.5K	10	37.5K
CIFAR-100 (ResNet)	50	12.5K	100	37.5K
CIFAR-100 (ViT)	80	20.0K	100	30.0K
ImageNet-100	50	31.9K	100	95.3K
CUB	100	1.5K	200	4.5K
Stanford Cars	98	2.0K	196	6.1K
FGVC-Aircraft	50	1.7K	100	5.0K
Herbarium 19	341	8.9K	683	25.4K

Table 12. A list of generic and fine-grained benchmarks used in the experiments. **#Image** and **#Class** indicate the number of samples and classes respectively.

B.8. Further Enhancement with SSL Methods.

In this section, we complement TRAILER with different SSL methods following OpenLDN [33]. In specific, such a pipeline first generates pseudo-labels on novel classes in the first training stage and then directly casts this task into traditional SSL with generated pseudo-labels to boost performance in the second stage of training. The second stage is compatible with different SSL approaches. Here we adopt Mixmatch [3], Fixmatch [34], and UDA [39]. Results in Table 9 further verify the feasibility of further enhancement of TRAILER with different traditional SSL methods. We further provide comparisons with OpenLDN when equipped with different SSL methods. As shown in Figure 11, TRAILER consistently outperforms OpenLDN when equipped with different traditional SSL methods.

B.9. Analysis of Training Complexity

We further provide the analysis of training complexity in this section. TRAILER does not impose much extra training complexity. (i)-It discards cumbersome pair-wise similarity calculation compared to other baselines of open-world SSL and frees the classifier h_{etf} from optimization. (ii)-Extra complexity mostly comes from h_{aux} and optimal transport (OT). h_{aux} is a linear layer and OT can be efficiently solved. Their training time is shown in Table 11, where they only occupy 5.9% and 1.0% of time.

B.10. Dataset Details

We conduct experiments of TRAILER on generic datasets following [4, 30, 33] and fine-grained datasets following previous protocol [37, 43]. The details of these datasets are listed in Table 12. Notably, the data split of labeled and unlabeled on CIFAR-100 is different for protocol [4, 30, 33] with ResNet and that [37] with ViT backbone.

Setting	Labeled $\mathcal{D}_l$ Known	Unlabeled $\mathcal{D}_u$ Known	Unlabeled $\mathcal{D}_u$ Novel (Unseen)
Supervised Learning	Yes	No	No
Traditional SSL	Yes	Yes	No
Robust/Open-set SSL	Yes	Yes	Yes (Reject)
Novel Class Discovery	Yes	No	Yes (Discovery)
Open-world SSL/GCD	Yes	Yes	Yes (Discovery)

Table 13. Comparisons with some related problem setting.

C. Additional Related Work

C.1. Literature of Positive-Unlabeled Learning

Positive-unlabeled learning (PU learning) [2, 12, 45] addresses an important scenario of binary classification where the training data only contains positive and unlabeled samples. Early approaches [19, 29, 40] for PU learning typically adopt the two-step heuristic pipeline, which first identifies reliable negative or positive samples from the unlabeled data and then trains a binary classifier with different semi-supervised learning manners for label assignment, such as graph-based [8, 46], confidence-based [31], and generative-based methods [23]. In addition to two-step methods, PU learning tasks can also be tackled by one-step approaches [25, 35, 41]. Among them, cost-sensitive methods [28, 42] are widely adopted which maintain different importance weights for samples, and then construct different risk estimators for optimization. The risk estimators include unbiased risk estimators [14], convex unbiased risk estimators [15], non-negative risk estimators [26], and so on. Despite the success, most of these methods are established on a known class prior and heavily hinge on an accurate class prior estimation as the prerequisite.

A flurry of methods has also been recently designed for PU learning without a class prior [9, 24, 27, 44]. CAPU [7] jointly estimates the class prior and learns a classifier. A novel mixup regularization technique is proposed in [27]. VPU [9] leverages the variational principle for boosted performance. PAN [24] modifies the generator architecture of GAN into a classifier to learn from PU data. Among these PU methods, we choose an empirically strong and easy-to-use VPU [9] algorithm for the known-novel separation in the label refinery step of our TRAILER framework.

C.2. Comparisons with Related Settings

We further provide the comparisons for open-world semi-supervised learning with some other related settings in Table 13, including our focused open-world SSL (also known as generalized category discovery and dubbed as GCD), supervised learning, traditional SSL, robust SSL (similar to open-set SSL), and novel class discovery. Please also refer to the pioneering work ORCA [4] for more details.

D. Overall Algorithm

We describe the overall training pipeline of our proposed TRAILER in Algorithm 1.

Algorithm 1 Pipeline of TRAILER

```
Input: Labeled set $\mathcal{D}_l = \{x_i, y_i\}_{i=1}^m$, unlabeled set $\mathcal{D}_u = \{x_i\}_{i=1}^n$ with $K = |\mathcal{C}_{all}|$; model $f_{\theta,\phi} = h_\phi \circ g_\theta$ with $\phi^{etf}$ fixed as a random simplex EFT;
# Warm-up phase
Warm-Up($\mathcal{D}, \theta, \phi$) with $\mathcal{L}_{warm}$ in Eq. (1)
for $epoch = 1, 2, ...,$ do
    Induce representation $z_i = g(x_i)$
    # Hierarchical Sample-Target Allocation
    for each unlabeled sample $x_i \in \mathcal{D}_u$ do
        # Rough assignment with optimal transport
        $q_i =$ Sinkhorn($h_{etf}(z_i)$) as Eq. (7)
        # Label refinery with known-novel separation
        Optimize $h_{aux}$ with $\mathcal{L}_{aux}$ and generate $\hat{y}^{aux}$
        $q^j = q^j$ if $\mathbb{I}(j \in \mathcal{C}_{known}) = \hat{y}^{aux}$ else 0
        Hard pseudo-label $\hat{y} = \arg\max_{1 \le j \le K} q^j$
    end for
    # Pseudo-label filtering class by class
    for each class $j = 1, 2, ..., K$ do
        $\mathcal{D}_u^j = \{(x, \hat{y}) \in \mathcal{D}_u | \hat{y} = j\}$
        $\mathcal{D}_{sel}^j = \{(x, \hat{y}) \in \mathcal{D}_u^j | \, rank(q^j) < R\%\}$
    end for
    The integral selected set $\mathcal{D}_{sel} = \cup_{j=1}^K \mathcal{D}_{sel}^j$
    # Progressive representation alignment
    Calculate $\mathcal{L}_{etf}$ with targeted classifier as Eq. (4)
    Calculate prototypical loss $\mathcal{L}_{proto}$ as Eq. (5)
    Calibration with $\mathcal{L}_{cls} = \lambda \mathcal{L}_{proto} + (1 - \lambda) \mathcal{L}_{etf}$
    # Overall training objectives
    Minimize $\mathcal{L}_{total} = \mathcal{L}_{cls} + \mathcal{L}_{aux} + \mathcal{L}_{con} + \alpha \mathcal{L}_{ent}$
end for
```

References

[1] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*. OpenReview.net, 2020. 1

[2] Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: a survey. *Mach. Learn.*, 109(4):719–760, 2020. 6

[3] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, pages 5050–5060, 2019. 6

[4] Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning. In *ICLR*. OpenReview.net, 2022. 2, 3, 5, 6

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020. 1

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9630–9640. IEEE, 2021. 4

[7] Shizhen Chang, Bo Du, and Liangpei Zhang. Positive unlabeled learning with class-prior approximation. In *IJCAI*, pages 2014–2021. ijcai.org, 2020. 6

[8] Sneha Chaudhari and Shirish K. Shevade. Learning from positive and unlabelled examples using maximum margin clustering. In *ICONIP*, pages 465–473. Springer, 2012. 6

[9] Hui Chen, Fangqing Liu, Yin Wang, Liyue Zhao, and Hao Wu. A variational approach for learning from positive and unlabeled data. In *NeurIPS 2020*, 2020. 2, 3, 4, 5, 6

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607. PMLR, 2020. 2

[11] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, pages 2292–2300, 2013. 1

[12] François Denis. PAC learning from positive statistical queries. In *Algorithmic Learning Theory, 9th International Conference, ALT '98, Otzenhausen, Germany, October 8-10, 1998, Proceedings*, pages 112–126. Springer, 1998. 6

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*. OpenReview.net, 2021. 4

[14] Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. In *NeurIPS*, pages 703–711, 2014. 6

[15] Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *ICML*, pages 1386–1394. JMLR.org, 2015. 6

[16] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, pages 9568–9577. IEEE, 2021. 3

[17] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *ICCV*, pages 9264–9272. IEEE, 2021. 1, 2, 3

[18] Peifeng Gao, Qianqian Xu, Peisong Wen, Zhiyong Yang, Huiyang Shao, and Qingming Huang. Feature directions matter: Long-tailed learning via rotated balanced representation. In *ICLR*, pages 27542–27563. PMLR, 2023. 5

[19] Tieliang Gong, Guangtao Wang, Jieping Ye, Zongben Xu, and Ming Lin. Margin based PU learning. In *AAAI*, pages 3037–3044. AAAI Press, 2018. 6

[20] Lan-Zhe Guo, Zhenyu Zhang, Yuan Jiang, Yu-Feng Li, and Zhi-Hua Zhou. Safe deep semi-supervised learning for unseen-class unlabeled data. In *ICML*, pages 3897–3906. PMLR, 2020. 2

[21] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *ICCV*, pages 8400–8408. IEEE, 2019. 2

[22] Kai Han, Sylvestre-Alvise Rebuffi, Sébastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *ICLR*. OpenReview.net, 2020. 2, 3

[23] Yu-Guan Hsieh, Gang Niu, and Masashi Sugiyama. Classification from positive, unlabeled and biased negative data. In *ICML*, pages 2820–2829. PMLR, 2019. 6

[24] Wenpeng Hu, Ran Le, Bing Liu, Feng Ji, Jinwen Ma, Dongyan Zhao, and Rui Yan. Predictive adversarial learning from positive and unlabeled data. In *AAAI*, pages 7806–7814. AAAI Press, 2021. 6

[25] Masahiro Kato, Takeshi Teshima, and Junya Honda. Learning from positive and unlabeled data with a selection bias. In *ICLR*. OpenReview.net, 2019. 6

[26] Ryuichi Kiryo, Gang Niu, Marthinus Christoffel du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *NeurIPS*, pages 1675–1685, 2017. 3, 5, 6

[27] Changchun Li, Ximing Li, Lei Feng, and Jihong Ouyang. Who is your right mixup partner in positive and unlabeled learning. In *ICLR*. OpenReview.net, 2022. 6

[28] Wenkai Li, Qinghua Guo, and Charles Elkan. One-class remote sensing classification from positive and unlabeled background data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 14:730–746, 2021. 6

[29] Bing Liu, Wee Sun Lee, Philip S. Yu, and Xiaoli Li. Partially supervised classification of text documents. In *ICML*, pages 387–394. Morgan Kaufmann, 2002. 6

[30] Jiaming Liu, Yangqiming Wang, Tongze Zhang, Yulu Fan, Qinli Yang, and Junming Shao. Open-world semi-supervised novel class discovery. In *IJCAI*, pages 4002–4010. ijcai.org, 2023. 2, 6

[31] Curtis G. Northcutt, Tailin Wu, and Isaac L. Chuang. Learning with confident examples: Rank pruning for robust classification with noisy labels. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*. AUAI Press, 2017. 6

[32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019. 4

[33] Mamshad Nayeem Rizve, Navid Kardan, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Openldn: Learning to discover novel classes for open-world semi-supervised learning. In *ECCV*, pages 382–401. Springer, 2022. 2, 5, 6

[34] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020. 2, 6

[35] Guangxin Su, Weitong Chen, and Miao Xu. Positive-unlabeled learning from imbalanced data. In *IJCAI*, pages 2995–3001. ijcai.org, 2021. 6

[36] Yiyou Sun and Yixuan Li. Opencon: Open-world contrastive learning. *Trans. Mach. Learn. Res.*, 2023, 2023. 4

[37] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *CVPR*, pages 7482–7491. IEEE, 2022. 2, 3, 4, 5, 6

[38] Xin Wen, Bingchen Zhao, and Xiaojuan Qi. Parametric classification for generalized category discovery: A baseline study. In *ICCV*, pages 16544–16554. IEEE, 2023. 4

[39] Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *NeurIPS*, 2020. 6

[40] Hwanjo Yu, Jiawei Han, and Kevin Chen-Chuan Chang. PEBL: web page classification without negative examples. *IEEE Trans. Knowl. Data Eng.*, 16(1):70–81, 2004. 6

[41] Chuang Zhang, Dexin Ren, Tongliang Liu, Jian Yang, and Chen Gong. Positive and unlabeled learning with label disambiguation. In *IJCAI*, pages 4250–4256. ijcai.org, 2019. 6

[42] Yan Zhang, Tianjiao Yang, and Chenguang Zhang. A regularization-based positive and unlabeled learning algorithm for one-class classification of remote sensing data. In *Cyberspace Safety and Security - 12th International Symposium, CSS 2020, Haikou, China, December 1-3, 2020, Proceedings*, pages 172–183. Springer, 2020. 6

[43] Bingchen Zhao, Xin Wen, and Kai Han. Learning semi-supervised gaussian mixture models for generalized category discovery. In *ICCV*, pages 16623–16633, 2023. 3, 6

[44] Hengwei Zhao, Xinyu Wang, Jingtao Li, and Yanfei Zhong. Class prior-free positive-unlabeled learning with taylor variational loss for hyperspectral remote sensing imagery. *CoRR*, abs/2308.15081, 2023. 6

[45] Yunrui Zhao, Qianqian Xu, Yangbangyan Jiang, Peisong Wen, and Qingming Huang. Dist-pu: Positive-unlabeled learning from a label distribution perspective. In *CVPR*, pages 14441–14450. IEEE, 2022. 3, 5, 6

[46] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *NeurIPS*, pages 321–328. MIT Press, 2003. 6