Novel Class Discovery for Long-tailed Recognition

Anonymous authors
Paper under double-blind review

Abstract

While the novel class discovery has achieved great success, existing methods usually evaluate their algorithms on balanced datasets. However, in real-world visual recognition tasks, the class distribution of a dataset is often long-tailed, making it challenging to apply those methods. In this paper, we propose a more realistic setting for novel class discovery where the distribution of novel and known classes is long-tailed. The challenge of this new problem is to discover novel classes with the help of known classes under an imbalanced class scenario. To discover imbalanced novel classes efficiently, we propose an adaptive self-labeling strategy based on an equiangular prototype representation. Our method infers better pseudo-labels for the novel classes by solving a relaxed optimal transport problem and effectively mitigates the biases in learning the known and novel classes. The extensive results on CIFAR100, ImageNet100, and the challenging Herbarium19 datasets demonstrate the superiority of our method.

1 Introduction

Novel Class Discovery (NCD) has attracted increasing attention in recent years (Han et al., 2021; Fini et al., 2021; Vaze et al., 2022), which aims to learn novel classes from unlabeled data with the help of known classes. Despite the existing methods have achieved significant progress, they typically assume the class distribution is balanced, focusing on evaluations of balanced datasets. This setting, however, is less practical in realistic scenarios, where the class distributions are mostly long-tailed. To address this limitation, we advocate a more realistic NCD setting in this work, in which both known and novel-class data are long-tailed. Such a NCD problem setting is important, as it bridges the gap between the typical novel class discovery problem and the real-world applications, and remains challenging, as it is often difficult to learn long-tailed known classes, let alone discovering imbalanced novel classes jointly.

Most existing NCD methods have difficulty coping with the imbalanced class scenario due to their restrictive assumptions. In particular, the pairwise learning strategy (Han et al., 2021; Zhong et al., 2021b) often learns a poor representation for the tail classes due to insufficient positive pairs from tail classes. The more recent self-labeling methods (Asano et al., 2020; Fini et al., 2021) typically assume that the unknown class sizes are evenly distributed, resulting in misclassifying the majority class samples into the minority classes. An alternative strategy is to combine the typical novel class discovery method with the supervised long-tailed learning method (Zhang et al., 2021b; Menon et al., 2020; Kang et al., 2020; Zhang et al., 2021a). They usually need to estimate the novel-class distribution for post-processing or retraining the classifier. However, as our preliminary study shows (c.f. Tab.2), such a two-stage method suffers from inferior performance due to the noisy estimation of the distribution.

To address the aforementioned limitations, we propose a novel adaptive self-labeling learning framework to solve novel class discovery for long-tailed recognition. Our main idea is to generate a better pseudo-label for unseen classes, which enables us to alleviate biased learning under severe class imbalance. To this end, we develop a new formulation for pseudo-label generation process based on a relaxed optimal transport problem, which assigns pseudo labels to the novel-class data in an adaptive manner and partially alleviates the class bias by implicitly rebalancing the classes. Moreover, leveraging our adaptive self-labeling strategy, we extend the equiangular prototype-based classifier (Yang et al., 2022b) to the imbalanced novel class clustering, which further mitigates long-tailed learning of known and novel classes in a unified manner.

Specifically, our model consists of an encoder with unsupervised pretraining and an equiangular prototype-based classifier. Given a batch of known and novel-class data, we encode those data into a unified embedding space. To learn from the known classes, we minimize the distance between each data embedding and its corresponding class prototype. Both the unsupervisedly-pretrained encoder and the equiangular prototypes help mitigate the imbalanced learning of the known classes. To learn the novel classes, we develop a novel adaptive self-labeling loss, which formulates the class discovery as a relaxed Optimal Transport problem, which is solved by an efficient bi-level optimization algorithm, and can be jointly optimized with the supervised loss of the known classes. Moreover, we design an efficient iterative learning algorithm that alternates between generating soft pseudo-labels for the novel-class data and performing class representation learning. In such a strategy, the learning bias of novel classes can be significantly reduced by our equiangular prototype design and soft adaptive self-labeling learning. Additionally, we propose a novel method to estimate the number of novel classes under an imbalance scenario. This enables our method to be applicable in real-world scenarios with unknown numbers of novel classes.

We conduct extensive experiments on the constructed long-tailed dataset, CIFAR100 and Imagenet100, and two challenging natural long-tailed dataset, Herbarium19 and iNaturalist18. The results demonstrate the efficiency of our proposed method. To summarize, the main contributions of our works are four-folds:

- We present a more realistic novel class discovery setting, where the known and novel classes are long-tailed.
- We introduce a novel adaptive self-labeling learning framework that generates pseudo labels of novel class in an adaptive manner and extends the equiangular prototype-based classifier to address the challenge of imbalanced novel class clustering.
- We formulate imbalanced novel class clustering as a relaxed optimal transport problem and develop a bi-level optimization strategy.
- We conduct extensive experiments on several benchmarks with different settings. And the sizeable improvement validates the effectiveness of our method.

2 Related Work

Novel class discovery Novel Class Discovery (NCD) aims to automatically learn novel classes in the open world when given knowledge of known classes. It typically assumes a semantic relation between novel classes and known classes, and the knowledge learned from known classes enables the model to better cluster novel classes. The associated deep learning problem was introduced in (Han et al., 2019), and the subsequent works can be grouped into two categories based on the learning objective they adopt to discover novel classes. One category of methods (Han et al., 2021; Zhong et al., 2021a;b; Hsu et al., 2018a;b) assume neighbouring samples in representation space belong to the same semantic category with high-probability. Based on this assumption, they learn a representation by minimizing the distances between adjacent data and maximizing non-adjacent ones, which is then used to group unlabeled data into novel classes. The other category of methods (Fini et al., 2021; Yang et al., 2022a) adopt a self-labeling technique. They assume that novel classes are equally sized, utilize the optimal transport-based self-labeling (Asano et al., 2020) method to assign cluster labels to novel class samples, and then self-train the model with the generated pseudo label.

Although the above methods have achieved significant improvement, they usually adopt the setting that the class distribution of novel classes is uniform, which is often restrictive for real-world problems. For visual recognition tasks, the class distribution is typically long-tailed, making it more challenging to discover novel classes. Especially for the pair-wise objective-based method, the learning of tail classes could be better due to insufficient samples for the tail classes. For self-labeling-based methods, the head classes are often misclassified as the tail classes due to the restrictive uniform distribution constraint. Moreover, both two methods tend to learn a biased classifier under the long-tailed setting. In contrast, we propose an adaptive self-labeling learning framework that generates a high-quality pseudo label for the novel classes by solving a relaxed optimal transport problem. We also mitigate imbalanced learning of the classifier by adopting equiangular prototypes.

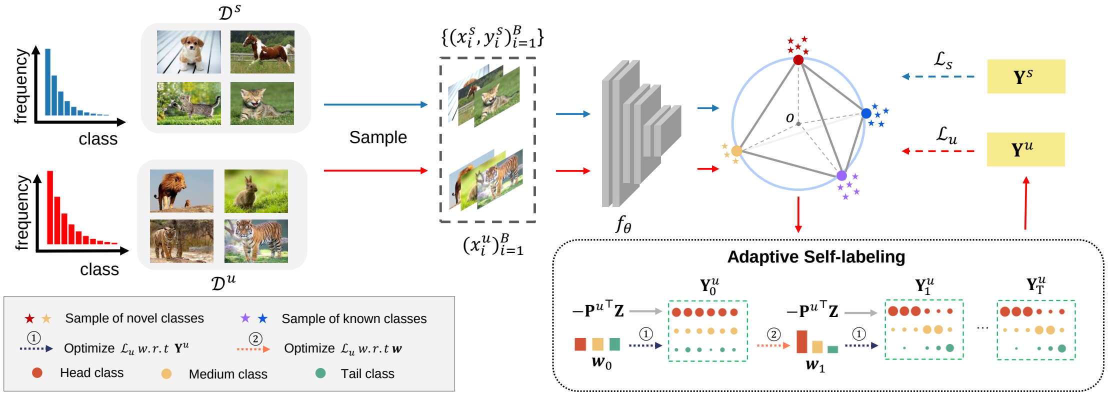

Figure 1: The overview of our framework. Our method first samples a batch including known and novel classes from the long-tailed dataset. And we use MSE loss to minimize the distance between samples and the equiangular prototype for known and novel classes. For novel classes, we propose a novel adaptive self-labeling method to assign the pseudo label, which is formulated as a bi-level optimization problem. Specially at step 1, we optimize $\mathcal{L}_u$ $w.r.t$ $\mathbf{Y}^u$. At step 2, we optimize $\mathcal{L}_u$ $w.r.t$ $\mathbf{w}$. This process is repeated in a loop until convergence. The optimization details are in Sec.4.3.

Supervised Long-tailed learning The supervised long-tailed learning aims to learn from labelled long-tailed data and perform well on both head and tail classes, e.g., (Zhang et al., 2021b; Weng et al., 2021). The core idea of those methods is to enhance the learning of the tail classes. One stream of strategies is to oversample tail classes on data (Han et al., 2005) or loss function (Cao et al., 2019) and learn a representation and classifier simultaneously. Among them, Logit-Adjustment (LA) (Menon et al., 2020) is a simple and effective method that has been widely used. In particular, LA mitigates the classifier bias by adjusting the logit based on class frequency in or after the learning. The other stream decouples the learning of representation and classifier (Kang et al., 2020; Zhang et al., 2021a). Especially, classifier retraining (cRT) (Kang et al., 2020) first learns a representation by instance-balanced resampling and then only retrains the classifier with the re-balanced technique. Although those methods have succeeded in supervised image recognition, applying them to novel classes where the distribution is unknown is difficult.

Recently, neural collapse (Papyan et al., 2020) demonstrates that the classifier vectors tend to converge to the vertices of a simplex equiangular tight frame (ETF). Inspired by this, Yang et al. (2022b) propose to initialize the classifier as ETF, and fix the parameter of the classifier during the learning, which helps to mitigate the classifier bias towards majority classes. The problem of classifier bias becomes even more severe in clustering imbalanced novel data, where both the representation and classifier are learned without clean label information. However, directly extending the ETF classifier to handle novel class discovery is infeasible due to the absence of ground-truth for novel classes. To this end, we leverage our adaptive self-labeling algorithm, and extend the ETF classifier to handle both known and novel classes, mitigating the imbalance learning of known and novel classes in a unified manner.

3 Problem Setup and Method Overview

We consider the problem of Novel Class Discovery (NCD) for visual recognition in a realistic setting, where the distribution of known and novel classes is typically long-tailed. In particular, we aim to learn a set of known classes $\mathcal{Y}^s$ from an annotated dataset $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^N$, and to discover a set of novel classes $\mathcal{Y}^u$ from an unlabeled dataset $\mathcal{D}^u = \{x_i^u\}_{i=1}^M$. Here $x_i^s, x_i^u \in \mathcal{X}$ are the input images and y_i^s are the known class labels in $\mathcal{D}^s$. For the NCD task, those two class sets have no overlap, i.e., $\mathcal{Y}^s \bigcap \mathcal{Y}^u = \emptyset$, and we denote their sizes as K^s and K^u respectively. In imbalance scenario, the numbers of training examples in different classes are imbalanced. For simplicity of notation, we assume that the known and novel classes are sorted by the cardinality of their training set in descending order. Specifically, we denote the number of training data for the known class i and the novel class j as N_i and M_j, accordingly, and we have $N_1 > N_2 \cdots > N_{K^s}, M_1 > M_2 \cdots > M_{K^u}$. To measure the class imbalance, we define an imbalance ratio for the known and novel classes, denoted as $R^s = \frac{N_1}{N_{K^s}}$ and $R^u = \frac{M_1}{M_{K^u}}$, respectively, where typically both $R^s, R^u \gg 1$.

To tackle the NCD task for long-tailed recognition, we propose a novel adaptive self-labeling framework capable of better learning both known and novel visual classes under severe class imbalance. Our framework consists of three key ingredients that help alleviate the imbalance learning of known and novel classes: 1) We introduce a classifier design based on equiangular prototypes for both known and novel classes, which mitigates class bias due to its fixed parametrization; 2) For the novel classes, we develop a new adaptive self-labeling loss, which formulates the class discovery as a relaxed Optimal Transport problem and can be jointly optimized with the supervised loss of the known classes; 3) We design an efficient iterative learning algorithm that alternates between generating soft pseudo-labels for the novel-class data and performing representation learning. An overview of our framework is illustrated in Fig.1. In addition, we propose a simple method to estimate the number of novel class in the imbalance scenario. The details of our method will be introduced in Sec. 4.

4 Our Method

In this section, we first introduce our model architecture and class representations in Sec. 4.1, followed by the loss design in Sec. 4.2 and Sec. 4.3 for the known and novel classes, respectively. Then, we summarize our iterative self-labeling learning algorithm in Sec. 4.4. Finally, we illustrate our proposed method to estimate the number of novel classes under imbalance scenario in Sec. 4.5.

4.1 Model Architecture and Class Representation

We adopt a generic design for the image classifier, consisting of an image encoder and a classification head for known and novel classes. Given an input x, our encoder network, denoted as f_θ, computes a feature embedding $\mathbf{z} = f_\theta(x) \in \mathbb{R}^{D\times 1}$, which is then fed into the classification head for class prediction. Here we normalize the feature embedding such that $\|\mathbf{z}\|_2 = 1$. While any image encoder can be potentially used in our framework, we adopt an unsupervised pretrained ViT model (Dosovitskiy et al., 2021) as our initial encoder in this work, which can extract a discriminative representation robust to the imbalanced learning (Liu et al., 2022). We also share the encoder of known and novel classes to encourage knowledge transfer between two class sets during model learning.

For the classification head, we consider a prototype-based class representation where each class i is represented by a unit vector $\mathbf{p}_i \in \mathbb{R}^{D\times 1}$. More specifically, we denote the class prototypes of the known classes as $\mathbf{P}^s = [\mathbf{p}^s_1, \cdots, \mathbf{p}^s_{K^s}]$ and those of the novel classes as $\mathbf{P}^u = [\mathbf{p}^u_1, \cdots, \mathbf{p}^u_{K^u}]$. The entire class space is then represented as $\mathbf{P} = [\mathbf{P}^s, \mathbf{P}^u] \in \mathbb{R}^{D\times(K^s+K^u)}$. To perform classification, given a feature embedding $\mathbf{z}$, we take the class of its nearest neighbour in the class prototypes $\mathbf{P}$ as follows,

$$c^* = \arg\min_i \ \|\mathbf{z} - \mathbf{P}_i\|_2 \tag{1}$$

where $\mathbf{P}_i$ is the i-th column of $\mathbf{P}$, and c^* is the predicted class of the input x.

In the imbalanced class scenario, it is typically challenging to learn the prototypes from the data as they tend to bias toward the majority classes, in particularly for the classifier learning of novel classes, where the classifier and representation are learned without label information. While many calibration strategies have been developed for the long-tailed problems in supervised learning (c.f. Sec. 2), they are not applicable to the imbalanced novel class discovery task as the label distribution of novel classes is unknown. To address this, we adopt a fixed parameterization for the class prototypes. Specifically, by leveraging pseudo label of novel data generated by our adaptive self-labeling algorithm (Sec.4.3) and the groundtruth of known data, we extend the method proposed by Yang et al. (2022b) in imbalance supervised learning scenario. And we utilize the vertices of a simplex equiangular tight frame (ETF) as the prototype of both known and novel classes, which alleviates the problem of biased prototype learning. Formally, our equiangular prototype $\mathbf{P}$ is generated by:

$$\mathbf{P} = \sqrt{\frac{K}{K-1}}\mathbf{M}(\mathbf{I}_K - \frac{1}{K}\mathbf{1}_{K\times K}) \tag{2}$$

where $\mathbf{M}$ is an arbitrary orthonormal matrix, $\mathbf{I}_K$ is an diagnoal matrix, $\mathbf{1}$ denotes the all ones matrix and $K = K^s + K^u$ is the total number of class prototypes. The generated prototypes have unit l_2 norm and same pair-wise angle. Those properties treat all classes equally, thus alleviating the classifier learning bias in imbalanced scenario.

4.2 Loss for Known Classes

For the known classes, we simply use the Mean Square Error (MSE) loss, which minimizes the l_2 distance between the feature embedding of an input x_i^s and the class prototype of its groundtruth label y_i^s. Specifically, we adopt the average MSE loss on the subset of known classes $\mathcal{D}^s$ as follows,

$$\mathcal{L}_s(\theta) = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{z}_i^s - \mathbf{p}_{y_i^s}^s\|_2 = -\frac{1}{N}\sum_{i=1}^{N} 2{\mathbf{z}_i^s}^\top \mathbf{p}_{y_i^s}^s + C \tag{3}$$

where $\mathbf{z}_i^s, \mathbf{p}_{y_i^s}^s$ is the feature embeddings and class prototypes, respectively, y_i^s is the groundtruth label of input x_i^s, and C is a constant. We note that our design copes with the imbalance in the known classes by adopting the equiangular prototype and initializing the encoder based on an unsupervised pretrained network, which is simple and effective (as shown in our experimental study)[1].

4.3 Adaptive Self-Labeling Loss for Novel Classes

We now present the loss function for discovering the novel classes in $\mathcal{D}^u$. Given an input x_i^u, we introduce a pseudo-label variable y_i^u to indicate its (unknown) membership to the K^u classes and define a clustering loss based on the Euclidean distance between its feature embedding $\mathbf{z}_i^u$ and the class prototypes $\mathbf{P}^u$ as follows,

$$\mathcal{L}_u(\theta) = \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{z}_i^u - \mathbf{p}_{y_i^u}^u\|_2 = -\frac{1}{M}\sum_{i=1}^{M} 2{\mathbf{z}_i^u}^\top \mathbf{p}_{y_i^u}^u + C \tag{4}$$

where C is a constant as the feature and prototype vectors are normalized. Our goal is to jointly infer an optimal membership assignment and learn a discriminative representation that better discovers novel classes.

Regularized Optimal Transport Formulation: Directly optimizing $\mathcal{L}_u$ is difficult, and a naive alternating optimization strategy often suffers from poor local minima (Caron et al., 2018), especially under the scenario of long-tailed class distribution. To tackle this, we reformulate the loss in Eq. 4 into a regularized Optimal Transport (OT) problem (Asano et al., 2020), which enables us to design an adaptive self-labeling learning strategy that iteratively generates high-quality pseudo-labels (or class memberships) and optimizes the feature representation jointly with the known classes. To this end, we introduce two relaxation techniques to convert the Eq. 4 to an OT problem as detailed below.

First, we consider a soft label $\mathbf{y}_i^u \in \mathbb{R}_+^{K^u}$ to encode the class membership of the datum x_i^u, where ${\mathbf{y}_i^u}^\top \mathbf{1}_{K^u} = 1$. Ignoring the constants in $\mathcal{L}_u$, we can re-write the loss function in a vector form as follows,

$$\min_{\mathbf{Y}^u} \mathcal{L}_u(\mathbf{Y}^u;\theta) = \min_{\mathbf{Y}^u} -\frac{1}{M}\sum_{i=1}^{M}\langle \mathbf{y}_i^u, {\mathbf{z}_i^u}^\top \mathbf{P}^u\rangle, \quad \text{s.t. } {\mathbf{y}_i^u}^\top \mathbf{1}_{K^u} = 1 \tag{5}$$

$$= \min_{\mathbf{Y}^u} \langle \mathbf{Y}^u, -{\mathbf{P}^u}^\top \mathbf{Z}\rangle_F, \quad \text{s.t. } \mathbf{Y}^u \mathbf{1}_{K^u} = \boldsymbol{\mu} \tag{6}$$

where $\langle,\rangle_F$ represents the Frobenius product, $\mathbf{Y}^u = \frac{1}{M}[\mathbf{y}_1^u, \cdots, \mathbf{y}_M^u]^\top \in \mathbb{R}_+^{M\times K^u}$ is the pseudo-label matrix, $\mathbf{Z} = [\mathbf{z}_1^u, \cdots, \mathbf{z}_M^u] \in \mathbb{R}^{D\times M}$ is the feature embedding matrix and $\boldsymbol{\mu} = \frac{1}{M}\mathbf{1}_M$. Such a soft label formulation is more robust to the noisy learning (Lukasik et al., 2020) as we use inferred pseudo-labels.

Second, as in (Asano et al., 2020), we introduce a constraint on the sizes of clusters to prevent a degenerate solution. Formally, we denote the cluster size distribution as a probability vector $\boldsymbol{\nu} \in \mathbb{R}_+^{K^u}$ and define the pseudo-label matrix constraint as ${\mathbf{Y}^u}^\top \mathbf{1}_M = \boldsymbol{\nu}$. Previous methods typically take an equal-size assumption

---

[1] While it is possible to integrate additional label balancing techniques, it is beyond the scope of this work.

Algorithm 1: Sinkhorn-Knopp Based Pseudo Labeling Algorithm

```
Input: Matrix $\mathbf{P}^{u\top}\mathbf{Z}$, marginal distribution $\boldsymbol{\mu}, \boldsymbol{\nu}$, hyperparameters $T, \lambda$
Output: $\mathbf{Y}$
Function Pseudo-Labeling($\mathbf{P}^{u\top}\mathbf{Z}, \boldsymbol{\mu}, \mathbf{w}$):
    $\mathbf{Y} \leftarrow \exp(\mathbf{P}^{u\top}\mathbf{Z}/\lambda)$
    $\mathbf{Y} \leftarrow \mathbf{Y}/\sum \mathbf{Y}$
    $\alpha, \beta \leftarrow \mathbf{1}, \mathbf{1}$
    for $t \in 1, 2, .., T$ do
        $\boldsymbol{\alpha} \leftarrow \boldsymbol{\mu}./(\mathbf{Y}\boldsymbol{\beta}), \boldsymbol{\beta} \leftarrow \mathbf{w}./(\mathbf{Y}^\top \boldsymbol{\alpha})$
    end
    $\mathbf{Y} \leftarrow diag(\boldsymbol{\alpha})\mathbf{Y}diag(\boldsymbol{\beta})$
    return $\mathbf{Y}$;
End Function
```

(Asano et al., 2020; Fini et al., 2021), where $\boldsymbol{\nu}$ is a uniform distribution. While such an assumption can partially alleviate the class bias by implicitly rebalancing the classes, it is often too restrictive for an unknown long-tailed class distribution. In particular, our preliminary empirical results show that it often forces the majority classes to be mis-clustered into minority classes, leading to noisy pseudo-label estimation. To remedy this, we propose a second relaxation mechanism on the above constraint. Specifically, we introduce an auxiliary variable $\mathbf{w} \in \mathbb{R}_+^{K^u}$, which is dynamically inferred during learning and encodes a proper constraint on the cluster size distribution, and formulate the loss into a regularized OT problem as follows:

$$\min_{\mathbf{Y}^u, \mathbf{w}} \mathcal{L}_u(\mathbf{Y}^u, \mathbf{w}; \theta) = \min_{\mathbf{Y}^u, \mathbf{w}} \langle \mathbf{Y}^u, -\mathbf{P}^{u\top}\mathbf{Z} \rangle_F + \gamma KL(\mathbf{w}, \boldsymbol{\nu}) \tag{7}$$

$$\text{s.t.} \quad \mathbf{Y}^u \in \{\mathbf{Y}^u \in \mathbb{R}_+^{M \times K^u} | \mathbf{Y}^u \mathbf{1}_{K^u} = \boldsymbol{\mu}, \mathbf{Y}^{u\top}\mathbf{1}_M = \mathbf{w}\} \tag{8}$$

where $\boldsymbol{\nu}$ is a probability vector and γ is the balance factor to adjust the strength of KL constraint in the second term. When $\gamma = \inf$, the KL constraint falls back to equality constraints. Intuitively, our relaxed optimal transport formulation allows us to generate better pseudo labels adaptively and alleviate the learning bias of head classes by proper label smoothing.

Pseudo Label Generation: Based on the regularized OT formulation of the clustering loss $\mathcal{L}_u$, we now present the pseudo label generation process when the encoder network f_θ is given. The generated pseudo labels will be used as the supervision of novel classes, which is combined with the loss of known classes for updating the encoder network. We will defer the overall training strategy to Sec. 4.4 and first describe the pseudo label generation algorithm below.

Eq. (7) and (8) minimize $\mathcal{L}_u$ *w.r.t* $(\mathbf{Y}^u, \mathbf{w})$ with a fixed cost matrix $-\mathbf{P}^{u\top}\mathbf{Z}$ (as θ is given). Instead of optimizing $\mathbf{Y}^u, \mathbf{w}$ directly by convex optimize techniques (Dvurechensky et al., 2018; Luo et al., 2023), which require a specific implementation and have unclear computational complexity in our scenario, we leverage the efficient Sinkhorn-Knopp algorithm (Cuturi, 2013) and propose a bi-level optimization algorithm to solve the problem approximately. Our approximate strategy consists of three main components as detailed below.

A. Alternating Optimization with Gradient Truncation: We adopt an alternating optimization strategy with truncated back-propagation (Shaban et al., 2019) to minimize the loss $\mathcal{L}_u(\mathbf{Y}^u, \mathbf{w})^2$. Specifically, we start from a fixed $\mathbf{w}$ (initialized by $\boldsymbol{\nu}$) and first minimize $\mathcal{L}_u(\mathbf{Y}^u, \mathbf{w})$ *w.r.t* $\mathbf{Y}^u$. As the KL constraint term vanishes, the task turns into a standard optimal transport problem, which can be efficiently solved by the Sinkhorn-Knopp Algorithm (Cuturi, 2013), as shown in Alg. 1. We truncate the iteration with a fixed T, which allows us to express the generated $\mathbf{Y}^u$ as a differentiable function of $\mathbf{w}$, denoted as $\mathbf{Y}^u(\mathbf{w})$. We then optimize $\mathcal{L}_u(\mathbf{Y}^u(\mathbf{w}), \mathbf{w})$ *w.r.t* $\mathbf{w}$ with simple gradient descent. The alternating optimization of $\mathbf{Y}^u$ and $\mathbf{w}$ takes several iterations to produce high-quality pseudo labels for the novel classes.

B. Parametric Cluster Size Constraint: Instead of representing the cluster size constraint $\mathbf{w}$ as a real-valued vector, we adopt a parametric function form in this work, which significantly reduces the search space of the optimization and typically leads to more stable optimization with better empirical results. Specifically, we

---

[2] Note that we simplify $\mathcal{L}_u(\mathbf{Y}^u, \mathbf{w}; \theta)$ to $\mathcal{L}_u(\mathbf{Y}^u, \mathbf{w})$ as we do not optimize θ in pseudo label generation process.

Algorithm 2: Adaptive Self-labeling Algorithm

```
Input: $\mathcal{D}^s, \mathcal{D}^u$, encoder $f_\theta$, equiangular prototype $\mathbf{P} \in \mathbb{R}^{D \times (K^s + K^u)}$,
    initial mini-batch Buffer, $\mathbf{w}, \boldsymbol{\mu} = \mathbf{1}_{J \times 1}$, hyperparameters $B, L$
for $e \in 1, 2, .., Epoch$ do
    for $s \in 1, 2, ..., Step$ do
        $\{(x_i^s, y_i^s)\}_{i=1}^B \leftarrow$ Sample($\mathcal{D}^s$), $\{x_i^u\}_{i=1}^B \leftarrow$ Sample($\mathcal{D}^u$)
        $\mathbf{z}^s = f_\theta(x^s), \mathbf{z}^u = f_\theta(x^u)$
        //MSE loss for labeled data
        $\mathcal{L}_s = \frac{1}{B} \sum_{i=1}^B ||\mathbf{z}_i^s - \mathbf{P}_{y_i^s}||^2$
        $\mathbf{y}^u = \mathbf{z}^{u\top} \mathbf{P}^u \in \mathbb{R}^{1 \times K^u}$
        $\mathbf{Y}^u = \text{Buffer}([\mathbf{y}_1^u; \mathbf{y}_2^u ..; \mathbf{y}_M^u]) \in \mathbb{R}^{J \times K^u}$
        for  $l \in 1, 2, ..., L$ do
            $\mathbf{Y}^u = \texttt{Pseudo-Labeling}(\mathbf{Y}^u, \boldsymbol{\mu}, \mathbf{w})$
            $\mathbf{w} \approx \arg\min_{\mathbf{w}} \mathcal{L}_u(\mathbf{Y}^u(\mathbf{w}), \mathbf{w})$
        end
        //MSE loss for unlabeled data
        $\mathcal{L}_u = \langle \mathbf{Y}^u, -\mathbf{P}^{u\top} \mathbf{Z} \rangle_F$
        minimize $\mathcal{L}_s + \alpha \mathcal{L}_u$ $w.r.t$ $\theta$
    end
end
```

parametrize $\mathbf{w}$ as a function of parameter τ in the follow form:

$$\mathbf{w}_i = \tau^{\frac{-i}{K^u - 1}}, \quad i = 0, 1, ..., K^u - 1 \tag{9}$$

Where τ can be viewed as the imbalance factor. As our class sizes decrease in our setting, we replace τ with a function form of $1 + \exp(\tau)$ in practice, which is always larger than 1. Then we normalize $\mathbf{w}_i$ by $\sum_{i=0}^{K^u - 1} \mathbf{w}_i$ to make it a valid probability vector.

C. Mini-Batch Buffer: We typically generate pseudo labels in a mini-batch mode (c.f. Sec. 4.4), which however results in unstable optimization of the OT problem. This is mainly caused by poor estimation of the cost matrix due to insufficient data, especially in the long-tailed setting. To address this, we build a mini-batch buffer to store $J = 2048$ history predictions (i.e., $\mathbf{P}^{u\top}\mathbf{Z}$) and replay the buffer to augment the batch-wise optimal transport computation. Empirically, we found that this mini-batch buffer significantly improves the performance of the novel classes.

4.4 Joint Model Learning

Given the loss function $\mathcal{L}_s$ and $\mathcal{L}_u$ for the known and novel classes, we develop an iterative learning procedure for the entire model. As our classifier prototypes are fixed, our joint model learning focuses on the feature representation learning, represented by the encoder network f_θ. Specifically, given the datasets of known and novel classes, $(\mathcal{D}^s, \mathcal{D}^u)$, we sample a mini-batch of known and novel classes data at each iteration, and perform the following two steps: 1) For novel-class data, we generate their pseudo labels by optimizing the regularized OT-based loss, as shown in Sec. 4.3; 2) Given the inferred pseudo labels for the novel-class data and the ground-truth labels for the known classes, we perform gradient descent on a combined loss function as follow,

$$\mathcal{L}(\theta) = \mathcal{L}_s(\theta) + \alpha \mathcal{L}_u(\theta), \tag{10}$$

where $\mathcal{L}_s$ is the loss for the known classes, $\mathcal{L}_u$ is the loss for the novel classes, and α is the factor to balance the learning of known and novel classes. The above learning process minimizes the overall loss function over the encoder parameters and pseudo labels in an alternative manner. Finally, we summarize the entire learning algorithm in Alg.2.

4.5 Estimation the number of novel categories

We propose a simple and effective method for estimating the number of novel classes, K^u, in an imbalanced scenario. Our approach involves an initial selection of K^u, followed by the use of a hierarchical clustering

Table 1: The details of each datasets. R^s is 50 for CIFAR100-50-50 and ImageNet100-50-50.

Datasets	**CIFAR100-50-50**		**ImageNet100-50-50**		**Herbarium19**	**iNaturalist18-1K**	**iNaturalist18-2K**
R^u	50	100	50	100	**UnKnown**	**UnKnown**	**UnKnown**
Known Classes	50	50	50	50	342	500	1000
Known Data	6.4k	6.4k	16.5k	16.5k	17.8k	26.3k	52.7k
Novel Classes	50	50	50	50	342	500	1000
Novel Data	6.4k	5.5k	16.2k	14.0k	16.5k	26.4k	49.9k

algorithm to cluster both known and novel classes ($\mathcal{D}^s, \mathcal{D}^u$). Next, we use the Hungarian algorithm to find the optimal mapping between the set of cluster indices and known class labels, and evaluate the performance of the known classes. Finally, we determine the best value of K^u by choosing the setting with the highest accuracy of the known classes (Vaze et al., 2022).

However, in imbalanced datasets, the average performance of known classes tends to be biased towards larger classes, which results in an underestimation of the number of estimated classes. To overcome this issue, we consider the average accuracy over class, which is not influenced by imbalance, and use a mixed metric to search for the optimal value of K^u. The mixed metric is defined as the weighted sum of the average accuracy of each sample, denoted by denoted by Acc_s, and each class on known classes, denoted by Acc_c, as follows:

$$Acc = \beta Acc_s + (1 - \beta) Acc_c, \tag{11}$$

where β is a weighting parameter and is set as 0.5 empirically. We employ the mixed metric to perform a binary search for the optimal value of K^u. The detail algorithm is shown in Appendix A.

5 Experiments

5.1 Experimental Setup

Datasets We evaluate the performance of our method on three datasets, including two long-tailed versions of image classification datasets, CIFAR100 (Krizhevsky et al., 2009) and ImageNet100 (Deng et al., 2009), and two real-world medium/large-scale long-tailed image classification datasets, Herbarium19 (Tan et al., 2019) and iNaturalist18 (Van Horn et al., 2018). To alleviate the challenge of clustering thousands of novel classes in imbalanced scenarios and reduce training costs, we subsample 1k and 2k classes from iNaturalist18 to create iNaturalist18-1K and iNaturalist18-2K, respectively. For all datasets, we randomly divide all classes into 50% known classes and 50% novel classes. For CIFAR100 and ImageNet100, we create "long-tailed" datasets for the known and novel classes by downsampling data examples per class following the exponential profile in (Cui et al., 2019) with imbalance ratio $R = \frac{N_1}{N_K}$. In order to explore the performance of novel class discovery under different scenarios, we set the imbalance ratio of known classes R^s as 50 and that of novel classes R^u as 50 and 100, which represent typical settings in long-tailed image recognition tasks. To report the class performance, we evaluate all methods on a balanced test dataset in each scenario and collect statistics on both known and novel classes. The details of each dataset are shown in Tab. 1.

Metric To evaluate the performance of our model on each dataset, we calculate the average accuracy over classes on test dataset. We measure the clustering class accuracy by comparing the hidden ground truth labels y_i with the model predictions $\hat{y}_i$ using the following formula:

$$\texttt{ClusterAcc} = \max_{perm \in P} \frac{1}{N} \sum_{i=1}^{N} y_i = perm(\hat{y}_i) \tag{12}$$

where P represents the set of all permutations and combinations. To optimize permutations, we use the Hungarian algorithm (Kuhn, 1955). It is important to note that we perform the Hungarian assignment for all categories only once, and then measure the classification class accuracy on both the known and novel subsets. We also sort the categories class according to their sizes in descending order and divide them into [Head: Medium: Tail] sections with the ratio of 3: 4: 3 for all datasets.

Table 2: Long-tailed novel class discovery performance on CIFAR-100, ImageNet100. We report average class accuracy on test datasets. R^s, R^u are the imbalance factors of known and novel classes respectively. "+LA" means post processing with logits-adjustment (Menon et al., 2020), "+cRT" means classifier retraining (Kang et al., 2020).

	CIFAR100-50-50						ImageNet100-50-50					
	$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$			$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$		
Method	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**
Autonovel	44.42	22.28	66.56	44.04	22.12	65.96	67.50	47.96	87.04	63.86	40.88	86.84
Autonovel + LA	45.32	20.76	69.88	42.20	18.00	66.40	67.74	46.72	**88.76**	64.08	39.32	**88.84**
AutoNovel + cRT	47.20	26.48	67.92	41.94	22.62	61.26	67.76	49.88	85.64	63.90	42.40	85.40
UNO	50.82	34.10	67.54	49.50	31.24	67.76	65.30	43.08	87.52	62.52	37.84	87.20
UNO + LA	52.36	33.82	**70.90**	51.62	31.04	**72.20**	65.92	43.12	88.72	63.28	37.72	88.84
UNO + cRT	**54.26**	40.42	68.10	47.62	31.02	64.22	68.38	50.80	85.96	63.10	39.96	86.24
Ours	53.75	**40.60**	66.90	**51.90**	**36.80**	67.00	**73.94**	**61.48**	86.40	**69.38**	**51.96**	86.80

Table 3: Long-tailed novel class discovery performance on medium/large-scale Herbarium19 and iNaturalist18. Other details are the same as Tab. 2.

	Herbarium			iNaturalist18-1K			iNaturalist18-2K		
Method	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**
Autonovel	34.58	9.96	59.30	42.33	11.67	73.00	39.08	8.57	69.60
Autonovel + LA	32.54	8.60	56.56	42.40	11.27	**73.53**	44.67	14.33	**75.00**
AutoNovel + cRT	45.05	24.46	64.49	44.20	16.13	72.27	37.95	9.27	66.63
UNO	47.47	34.50	60.58	52.93	31.60	74.27	45.60	19.97	71.23
UNO + LA	46.76	27.96	**65.69**	46.63	24.33	**74.60**	46.63	20.33	72.93
UNO + cRT	46.47	33.13	59.95	51.73	32.60	70.87	46.47	24.90	68.03
Ours	**49.21**	**36.93**	61.63	**58.87**	**45.47**	72.27	**49.57**	**34.13**	65.00

Implementation Details For fair comparisons, all methods use a ViT-B-16 backbone as the image encoder, which is pre-trained with DINO (Caron et al., 2021) in an unsupervised manner. For our method, we train 50 epochs on CIFAR100 and ImageNet100, 70 epochs on Herbarium. We use AdamW with momentum as the optimizer with linear warm-up and cosine annealing (lr_{base} = 1e-3, lr_{min} = 1e-4, and weight decay 5e-4). We set $\alpha = 1$, and select $\gamma = 500$ by validation set. In addition, we analyze the sensitivity of γ in Appendix C. For all experiments, we set the batch size to 128 and the iteration step L to 10. For the Sinkhorn-Knopp algorithm, we adopt all the hyperparameters from (Caron et al., 2020), e.g. $n_{iter} = 3$ and $\epsilon = 0.05$. Implementation details of other methods can be found in Appendix B.

5.2 Comparison with SOTA

Tab.2 shows a comparison of our method with other baselines on the CIFAR100, Imagenet100, and Herbarium19 datasets[3]. For CIFAR100, when $R^s = R^u = 50$, our method achieves results that are competitive compared to the two-stage training methods. As the data become more imbalanced, i.e. $R^u = 100$, our method achieves **5.78%** improvement on the novel class accuracy. We note that our method does not exhibit a significant advantage due to the limited quality of the representation computed from the low-resolution images. For ImageNet100, our method achieves large improvements in different R^u settings, surpassed the previous SOTA method by **10.68%** and **9.56%**.

Furthermore, in Tab. 3, we show the results on medium/large scale datasets. Specifically, on the challenging fine-grained imbalanced Herbarium19 dataset, which contains 341 known classes, our method also achieves **2.43%** improvement on the novel class accuracy compared to UNO. We also report the per-sample average class accuracy in Appendix E, on which we achieve $\sim$ 10% improvement. On the more challenging iNaturalist18-1k and iNaturalist18-2k datasets, we observe a significant improvement ($>$ 10%) in the performance of novel classes compared to the Herbarium19 dataset. In summary, our notable performance gains in multiple experimental settings demonstrate that our method is effective for the challenging long-tailed NCD task.

---

[3] More analysis of NCD methods with class re-balancing techniques is included in Appendix D.

Table 4: Estimation the number of novel categories. R^s is 50 for CIFAR100 and ImageNet100 datasets.

Method	CIFAR100-50-50		ImageNet100-50-50		Herbarium
	$R^u = 50$	$R^u = 100$	$R^u = 50$	$R^u = 100$	Unknown
GT	50	50	50	50	341
Baseline	0	10	7	14	2
Ours	20	29	59	59	153

Table 5: Experiments on three datasets when K^u is unknown.

	CIFAR100-50-50						ImageNet100-50-50						Herbarium		
	$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$			$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$			Unknown		
Method	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**
AutoNovel	41.25	16.08	66.42	43.74	17.64	69.84	63.92	37.80	90.04	66.68	44.96	88.40	37.84	15.64	60.16
AutoNovel+LA	41.82	16.18	67.46	43.82	18.08	69.56	61.74	32.68	90.80	62.26	35.52	89.00	42.15	19.06	65.37
AutoNovel+cRT	45.81	19.50	72.12	43.44	18.18	68.70	62.83	37.96	87.68	52.44	16.68	88.20	40.95	18.86	63.16
UNO	47.67	28.92	66.42	46.56	25.92	67.20	67.96	48.48	87.44	64.16	41.24	87.08	40.83	23.55	58.24
UNO+LA	49.51	28.18	70.84	48.02	25.70	70.34	67.94	47.44	88.44	65.02	41.48	88.56	42.83	23.02	62.56
UNO+cRT	49.35	30.82	67.88	45.49	26.68	64.30	70.94	55.32	86.56	62.76	38.72	86.80	40.67	22.84	58.63
Ours	49.03	32.66	65.40	48.89	33.06	64.72	74.06	61.04	87.08	68.94	50.84	87.04	44.20	29.01	59.34
Ours+LA	49.64	32.46	66.82	49.87	33.12	66.62	74.38	61.48	87.28	71.08	54.92	87.24	45.74	29.55	61.90

5.3 Estimation the number of novel categories

To evaluate the effectiveness of our estimation method, we establish a Baseline that uses the average accuracy as the indicator to search for the optimal K^u value by hierarchical clustering. The details of algorithm are shown in A. As shown in Tab.4, our proposed method significantly outperforms the Baseline on three datasets and various scenarios, indicating the superiority of our proposed mixed metric based estimation method in imbalanced scenarios. Furthermore, we conduct experiments on three datasets with estimated K^u. As Tab.5 shown, our method achieves sizeable improvements on the novel and overall class accuracy, except in the case of CIFAR when $R^u = 50$, which we achieve comparable results. On the CIFAR dataset, the baseline surpasses our method on the known classes, especially equipped with the LA or cRT technique, resulting in our method being slightly better or worse than existing methods in overall accuracy. When our method is equipped with LA, we can achieve better results. While on ImageNet100 and Herbarium19 datasets, our method surpasses the existing methods by a significant margin. For example, on ImageNet100 dataset when $R^u = 100$, ours outperforms the best baseline (AutoNovel) by 3.92% in overall accuracy and 5.88% in novel accuracy. Moreover, when our method is equipped with LA, the performance is further improved, with an increase of 4.4% in overall accuracy and 9.96% in novel accuracy.

It is important to note that the effect of estimated K^u differs based on its relationship with the ground truth value. When the estimated K^u is lower than the ground truth value, such like CIFAR100 and Herbarium19, the performance deteriorates compared to using the true K^u. This occurs because the estimated lower K^u leads to the mixing of some classes, especially medium and tail classes, resulting in degraded performance. When the estimated K^u is higher than the ground truth value, such like ImageNet100, using the estimated K^u leads to better results for UNO and comparable results for Ours. For UNO which assume equally sized distribution, larger K^u tend to assign the head classes to additional empty classes, reducing the noise caused by the mixing of head classes with medium and tail classes, and thereby improving the accuracy of medium and tail classes. While, our method dynamically adjusts the allocation ratio for novel classes, effectively suppressing the assignment of head classes to empty classes, which allows us to achieve comparable results. A more detailed analysis of this phenomenon can be found in Appendix F.

5.4 Ablation study

Component analysis: In Tab.6, we ablate the components in our method on Imagenet100 and report model performance by adding each core component of our method in isolation, which includes the equiangular prototype representation, adaptive self-labeling and the mini-batch buffer. As shown in the first and second rows of the Tab.6, the addition of min-batch buffer results in a 2% improvement compared to the baseline on novel class accuracy. By comparing the second and third rows, we can see that all sub-part class accuracy

Table 6: Ablation study on ImageNet100. "EP" stands for equiangular prototype and "ASL" stands for adaptive self-labeling.

	$R^s = 50, R^u = 50$				$R^s = 50, R^u = 100$			
Method	**Novel**	**Head**	**Medium**	**Tail**	**Novel**	**Head**	**Medium**	**Tail**
Baseline	43.08	44.93	49.20	33.07	37.84	49.73	43.10	18.93
+ Buffer	46.36	46.93	51.90	38.40	39.88	52.00	46.00	19.60
+ Buffer + EP	57.40	66.00	**59.70**	45.73	47.24	68.67	49.20	23.20
+ Buffer + EP + ASL	**61.48**	**77.47**	55.80	**53.07**	**51.96**	**77.47**	**54.20**	**23.47**

Table 7: The effects of different combinations of loss function and classifier. Results on ImageNet100, for $R^s = 50, R^u = 50$. cls is an abbreviation for classifier.

Method	**Novel**	**Head**	**Medium**	**Tail**
Learnable cls + CE loss	46.36	46.93	51.90	38.40
EP cls + CE loss	52.40	53.47	**66.10**	33.07
EP cls + MSE loss	**57.40**	**66.00**	59.70	**45.73**

has been improved, particularly for the head class accuracy. For instance, in the $R^u = 100$ setting, our method achieves 16.67% improvement on the head, 7.8% improvement on medium, 7.33% improvement on tail classes.This demonstrates that the equiangular prototype representation helps alleviate the imbalance learning of novel classes, and learn a discriminative representation for all novel classes. Comparing the third and last row, we show that adopting adaptive self-labeling greatly improves the tail and head class accuracy. For example, our method achieves 11.47% improvement on head, and 7.34% improvement on tail classes for $R^u = 50$. The results indicate that the uniform constraint on the distribution of clusters is not suitable for imbalance clustering, as it tends to misclassify head classes samples as tail classes. And the reason for the slightly worse performance for medium classes is that the uniform distribution constraint better approximates the true medium distribution. However, this constraint harms the performance of the head and tail, resulting in a nearly 4% decrease in the novel class accuracy. In conclusion, the overall results validate the effectiveness of our proposed components. Especially the equiangular prototype and adaptive self-labeling produce notable improvements.

Effect of Equiangular Prototype: In Sec.4, we utilize MSE loss to minimize the distance between sample and prototype. We explore the effects of different combinations of loss functions and classifiers. As Tab.7 shown, a learnable classifier with CE loss performs worse than EP cls + CE loss. We argue that the learned prototype tends to bias to head classes, and the learned representation is less discriminative. What's more, EP cls with MSE loss improve EP cls with CE loss by a large margin, especially for head and tail classes. In the early learning stage, the representation is relatively poor, and massive head class samples are allocated to tail classes because of uniform distribution constraints. While the gradient of CE loss is larger than MSE loss, resulting in the EP + CE loss fitting on the noise pseudo-label quickly.

Moreover, to better understand the effect of the equiangular prototype for novel class clustering, we visualize the feature space by t-SNE(Van der Maaten & Hinton, 2008) on the test set. As Fig.2 shown, the feature representations learned by the equiangular prototype are more tightly grouped and more evenly distributed interclass distances. However, learnable classifier results in several classes being entangled together.

Adaptive self-labeling: In this part, we validate the effectiveness of our design on $\mathbf{w}$. We set $\mathbf{w}$ as the uniform and ground-truth distribution and conduct experiments, respectively. Interestingly, as shown in the first two rows of Tab.8, setting w as a uniform prior achieves better performance, especially on medium and tail. We speculate that the uniform constraint smooths the pseudo label of head class samples, mitigating the bias learning of head classes, thus improving the results of medium and tail classes. In addition, we also try two ways to learn w with a prior constraint of uniform distribution. One way is to parameterize w as a real-valued vector, and the other is to use a parametric form as a function of τ. As shown in the last two rows of Tab.8, we find that optimizing a k-dimensional $\mathbf{w}$ is unstable and prone to assign overly large cluster

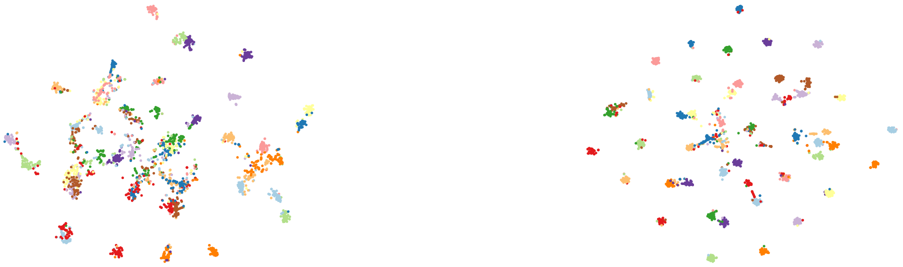

Figure 2: t-SNE visualization of novel instances in ImageNet100 for features after the last transform block. The left is the feature space using a learnable classifier, and the right is the feature space using equiangular prototype.

Table 8: The impact of different parameteric strategy of $\mathbf{w}$. The first two rows of $\mathbf{w}$ are fixed, and the last two rows represent the two parameterization ways of $\mathbf{w}$. Results on ImageNet100, for $R^s = 50, R^u = 50$.

Method	Novel	Head	Medium	Tail
$\mathbf{w}$ = Uniform	57.40	66.00	59.70	45.73
$\mathbf{w}$ = True Prior	42.40	64.53	46.90	14.27
$\mathbf{w}$	55.64	69.73	**59.80**	31.87
$\mathbf{w}(\tau)$	**61.48**	**77.47**	55.80	**53.07**

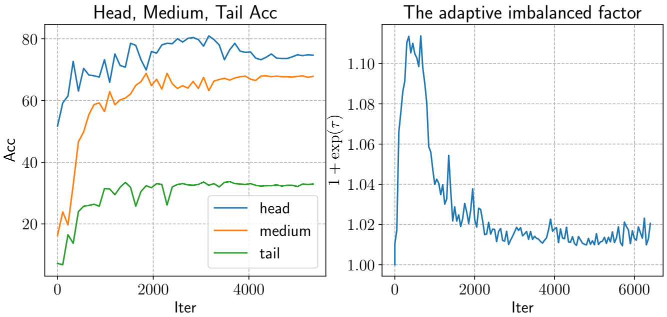

Figure 3: Analysis of $\mathbf{w}$ during training

sizes for some clusters. Therefore, optimizing $\mathbf{w}$ parametrized by a function of parameter τ (As shown in Eq.9) seems to be more effective.

To provide more analysis on $\mathbf{w}$, we visualize the learned imbalance factor and the head/medium/tail class accuracy during the training process of the model. As Fig.3 shows, in the early stage, the head class accuracy first increases quickly, indicating that the model bias on the head classes and the head representation is learned better. Correspondingly, the learned imbalance factor increase, thus assigning more samples to the head classes. Subsequently, the medium and tail class accuracy increase; meanwhile, the imbalance factor decreases, resulting in the pseudo label process bias to medium and tail classes. Although the imbalance factor changes a little during the learning, it improves novel class accuracy by a large margin compared to the fixed uniform prior (the first row and last row in Tab.8).

6 Conclusion

In this paper, we propose a realistic novel class discovery setting for image recognition, where known and novel classes are long-tailed. To assign pseudo labels to novel classes and mitigate imbalance learning, we propose a novel adaptive self-labeling framework, which formulates the pseudo-label assignment problem as a relaxed optimal transport problem and extends the equiangular prototype-based classifier to handle novel class discovery, effectively mitigating the challenges of imbalanced learning. Moreover, we propose an bi-level optimization algorithm to efficiently solve the relaxed optimal transport problem. We also propose a method to estimate the number of novel classes in an imbalanced scenario. Finally, we conduct massive experiments on two small-scale long-tailed CIFAR100 and ImageNet100 datasets, and two medium/large-scale real-world long-tailed Herbarium19 and iNaturalist18 datasets, demonstrating our method's superiority.

References

Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, 2019.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 132–149, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9268–9277, 2019.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

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. *ICLR*, 2021.

Pavel Dvurechensky, Alexander Gasnikov, and Alexey Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by sinkhorn's algorithm. In *International conference on machine learning*, pp. 1367–1376. PMLR, 2018.

Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9284–9292, 2021.

Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pp. 878–887. Springer, 2005.

Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8401–8409, 2019.

Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Autonovel: Automatically discovering and learning novel visual categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. In *International Conference on Learning Representations*, 2018a.

Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. Multi-class classification without multi-class labels. In *International Conference on Learning Representations*, 2018b.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *Eighth International Conference on Learning Representations (ICLR)*, 2020.

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

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2 (1-2):83–97, 1955.

Hong Liu, Jeff Z. HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=4AZz9osqrar`.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pp. 6448–6458. PMLR, 2020.

Yiling Luo, Yiling Xie, and Xiaoming Huo. Improved rate of first order algorithms for entropic optimal transport. In *International Conference on Artificial Intelligence and Statistics*, pp. 2723–2750. PMLR, 2023.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. Truncated back-propagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1723–1732. PMLR, 2019.

Kiat Chuan Tan, Yulong Liu, Barbara Ambrose, Melissa Tulig, and Serge Belongie. The herbarium challenge 2019 dataset. *arXiv preprint arXiv:1906.05372*, 2019.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778, 2018.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7492–7501, 2022.

Zhenzhen Weng, Mehmet Giray Ogut, Shai Limonchik, and Serena Yeung. Unsupervised discovery of the long-tail in instance segmentation using hierarchical self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2603–2612, 2021.

Muli Yang, Yuehua Zhu, Jiaping Yu, Aming Wu, and Cheng Deng. Divide and conquer: Compositional experts for generalized novel class discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14268–14277, 2022a.

Yibo Yang, Liang Xie, Shixiang Chen, Xiangtai Li, Zhouchen Lin, and Dacheng Tao. Do we really need a learnable classifier at the end of deep neural network? *arXiv preprint arXiv:2203.09081*, 2022b.

Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2361–2370, 2021a.

Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *arXiv preprint arXiv:2110.04596*, 2021b.

Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10867–10875, 2021a.

Zhun Zhong, Linchao Zhu, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Openmix: Reviving known knowledge for discovering novel visual categories in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9462–9470, 2021b.

A The algorithm of estimating the number of novel categories

We introduce a novel approach for estimating the number of unknown classes in imbalanced scenarios (Section 4.5). Our method leverages the clustering performance of the known classes dataset $\mathcal{D}^s$ as a means of searching for the optimal value of K^u. The detailed algorithm for this estimation process is outlined in Algorithm 3. In our method, we evaluate the performance using the mixed metric (Eqn. 11). In contrast, the "Baseline" method utilizes the average accuracy of each sample as its evaluation metric, which tends to bias to majority classes.

Algorithm 3: The algorithm of estimating the number of novel categories.

```
Input: $\mathcal{D}^s, \mathcal{D}^u, K^s$, maximum $K^u_{max}$, evaluation metric $eval$, hierarchical clustering algorithm $HC$
Output: $K^u_{mid}$
$K^u_{high}, K^u_{med}, K^u_{low} \leftarrow K^u_{max}, K^u_{max}//2, 0$
$Acc_{high} = eval(HC(\mathcal{D}^s, \mathcal{D}^u, K^u_{high} + K^s), \mathcal{D}^s)$
$Acc_{med} = eval(HC(\mathcal{D}^s, \mathcal{D}^u, K^u_{med} + K^s), \mathcal{D}^s)$
$Acc_{low} = eval(HC(\mathcal{D}^s, \mathcal{D}^u, K^u_{low} + K^s), \mathcal{D}^s)$
while $K^u_{high} > K^u_{low}$ do
    if $Acc_{high} > Acc_{low}$ then
        $K^u_{low} \leftarrow K^u_{med}$
        $K^u_{med} \leftarrow (K^u_{high} + K^u_{low})//2$
        $Acc_{low} = Acc_{med}$
        $Acc_{med} = eval(HC(\mathcal{D}^s, \mathcal{D}^u, K^u_{med} + K^s), \mathcal{D}^s)$
    end
    else
        $K^u_{high} \leftarrow K^u_{med}$
        $K^u_{med} \leftarrow (K^u_{high} + K^u_{low})//2$
        $Acc_{high} = Acc_{med}$
        $Acc_{med} = eval(HC(\mathcal{D}^s, \mathcal{D}^u, K^u_{med} + K^s), \mathcal{D}^s)$
    end
end
```

B Implementation details

As there are currently no existing baselines for novel class discovery in an imbalanced setting, we have implemented two typical NCD methods, AutoNovel (Han et al., 2021) and UNO (Fini et al., 2021). To handle imbalanced learning, we have combined these NCD methods with two common approaches for long-tailed problems: logit-adjustment (Menon et al., 2020) and decoupling the learning of representation and classifier head (Kang et al., 2020).

We have used the same unsupervised pretrained model and only modified the training setup of AutoNovel and UNO. Specifically, we trained AutoNovel for 200 epochs until convergence on all datasets, and the training strategy for UNO is identical to ours, as described in the main paper.

For our implementation of logit-adjustment, we have set $\pi = 1$ following (Menon et al., 2020). If the estimated number of pseudo-labels for a novel class is 0, we do not make any corrections to its logits. When adding cRT (Kang et al., 2020), we first estimate the class distribution and use the same number of epochs as in the first stage.

C Sensitive analysis of γ

The optimal value for the hyperparameter γ is selected by partitioning a subset of known classes as the validation set. Additionally, to investigate the sensitivity of the hyperparameter gamma, we have presented the change in novel class accuracy from gamma values of 100 to infinity in Figure 4. Our findings indicate that when gamma is larger than 300, our adaptive self-labeling method outperforms the naive baseline. However, the value selected using our validation set is not the optimal one.

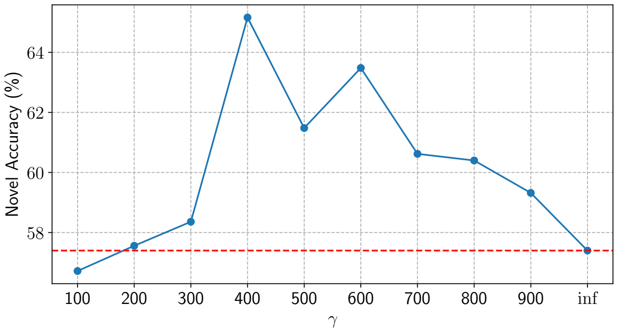

Figure 4: The sensitive analysis of γ

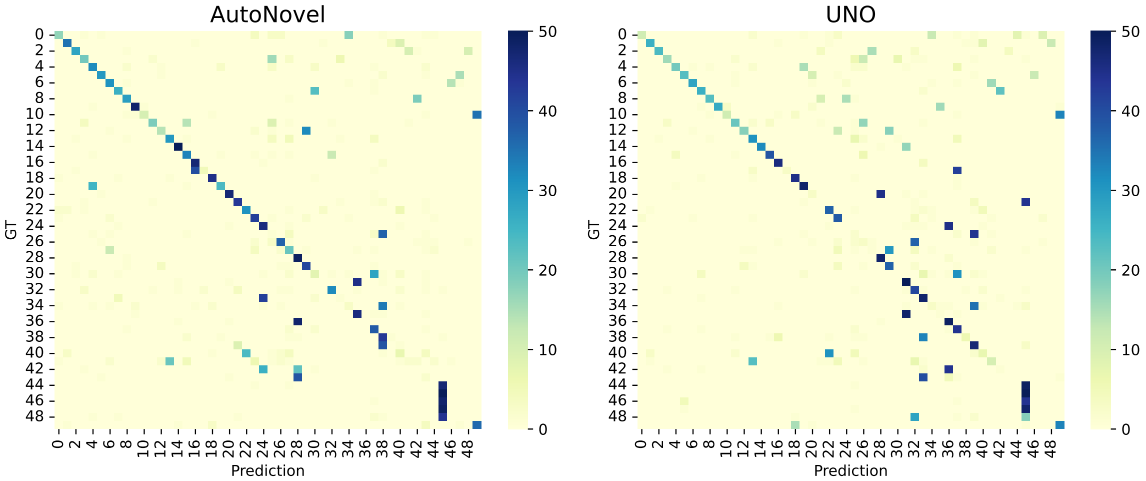

Figure 5: The confusion matrix of novel classes for typical NCD methods.

Table 9: The details analysis of typical NCD methods. Results on ImageNet100, for $R^s = 50, R^u = 50$.

Method	Novel	Head	Medium	Tail
Autonovel	47.96	55.87	55.60	29.87
UNO	43.08	44.93	49.50	32.13
Ours	**61.48**	**77.47**	**55.80**	**53.07**

Table 10: The results of UNO on the balance dataset.

Dataset	All	Novel	Known
CIFAR100	74.55	65.40	83.70
ImageNet100	85.12	76.96	93.28

D Analysis of NCD method

aragraphNCD methods on Head/Medium/Tail classes: In Tab. 2 of the main manuscripts, we have presented the results of NCD methods. To further analyze these methods, we have shown their performance on the Head, Medium, and Tail classes in the novel class in Tab. 9. Our proposed method shows an improvement of over 20% on both the Head and Tail classes, demonstrating its advantage.

Additionally, Autonovel performs worse on the Tail class due to the limited number of positive pair samples for tail classes. In contrast, UNO performs worse in the Head classes because the head classes are misclassified into Tail classes. This argument is supported by the confusion matrix shown in Fig. 5. Specifically, in the case of Autonovel, several tail classes are merged into a single class due to poor representation. There are too many samples on the right side of the confusion matrix for UNO, which denotes that the head classes are being misclassified into tail classes.

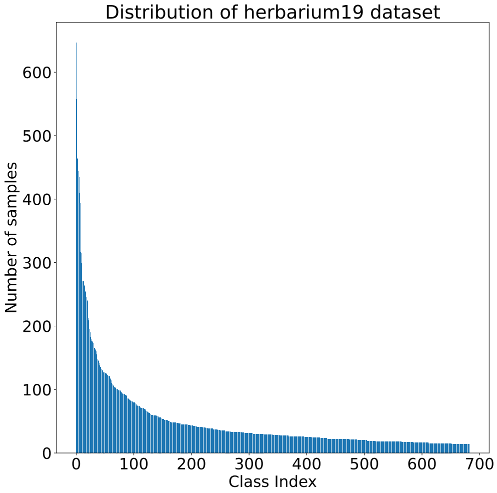

Figure 6: The distribution of herbarium19 dataset

NCD methods with class re-balancing techniques: We conclude that novel class discovery (NCD) for long-tailed data is challenging, and existing methods have not been able to solve this problem. As shown in Tab. 2 of the main paper and Tab.10, the novel class accuracy decreased by almost 30% on both CIFAR100 and ImageNet100 datasets.

To improve the performance of NCD in long-tailed scenarios, we have combined NCD with long-tail methods (+LA, +cRT). We observe that the accuracy of known and novel classes improves when the distribution estimation of novel classes is more accurate. Specifically, in CIFAR100, when $R^u = 50$, AutoNovel performs poorly in estimating the distribution of novel classes due to the use of pairwise loss, which assigns similar features the same pseudo label, making it difficult to learn distinctive representations for tail classes in an imbalanced setting. This results in tail classes being mixed with head classes. When AutoNovel is combined with long-tail methods, the novel classes decrease while UNO improves. When $R^u = 100$, the severe imbalance of novel classes makes learning novel classes more difficult, resulting in a worse estimated distribution. As a result, combining UNO with long-tail methods no longer has any effect.

On ImageNet, the estimated distribution of novel classes is more accurate, and both AutoNovel and UNO have improved the accuracy of novel class. However, UNO's accuracy of known classes slightly decreased because novel classes and known classes are often confused. For Herbarium19, the actual distribution is difficult to predict, so the achievement of LA and cRT is limited.

In conclusion, due to the noisy estimated distribution, naively combining NCD and long-tail methods cannot effectively solve the long-tailed novel class discovery problem.

E More results on Herbarium19 dataset

Fig.6 presents the distribution of the Herbarium19 dataset. The dataset is composed of 683 classes, out of which 178 categories have less than 20 samples, which presents a significant challenge when attempting to cluster novel classes. Tab.11 shows the average accuracy over both class and instance. Our proposed method outperforms the typical NCD methods by a considerable margin in both metrics, demonstrating the effectiveness of our approach.

F More explanation about estimation the number of novel categories

In order to better analyze the experiment of estimating the number of novel categories under a higher K^u, we visualize the novel class confusion matrices of UNO and Ours for known and unknown K^u with $R^s = R^u = 50$ in ImageNet100. The y-axis is groud-truth, and the x-axis is prediction.

Table 11: Long-tailed novel class discovery performance on Herbarium19. We report average class and samples accuracy on test datasets. The top three lines is the accuracy average over classes. The bottom three lines is the accuracy average over samples.

	Herbarium		
Method	**All**	**Novel**	**Known**
Autonovel	34.58	9.96	59.30
UNO	47.47	34.50	60.58
Ours	**49.21**	**36.93**	**61.63**
Autonovel	40.83	14.15	65.98
UNO	50.20	31.40	67.51
Ours	**55.66**	**41.15**	**69.33**

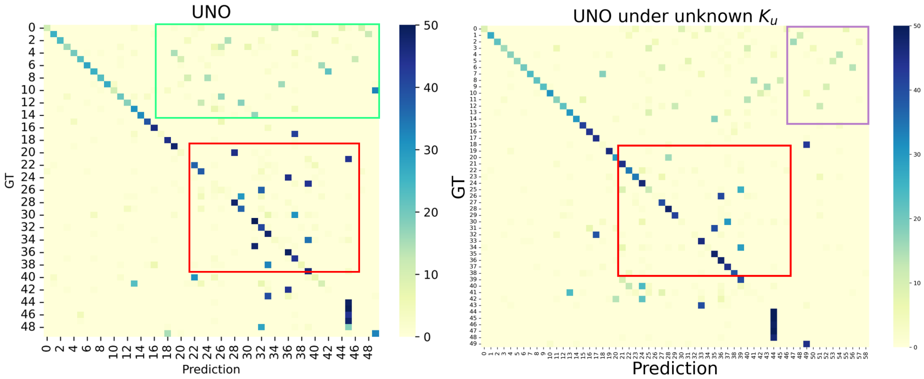

Figure 7: The confusion matrix of UNO for known and unknown K^u

UNO utilizes the Sinkhorn algorithm to generate pseudo labels, which enforces an equal distribution for each class. This will result in splitting a head class into multiple subcategories, which may be mixed up with the medium and tail classes, as illustrated by the green-bordered boxes in the left side image in Figure 7. Due to the misclassification of the head class, this will introduce noise that affects the quality of prediction for the medium and tail classes, as indicated by the red-bordered boxes in the left side image in Figure 7. When K^u is larger than the true K^u, this problem can be alleviated because the head class is more likely to be assigned to categories that do not match the ground truth, as shown by the purple-bordered boxes in the right diagram. In this case, the noise in the pseudo labels for the medium class and tail class will be reduced, thus improving the accuracy of medium and tail classes, as shown in the the red-bordered boxes in the right side image in Figure 7.

According to the result analysis, we find that our method's novel classes accuracy will not be significantly affected when K^u is greater than the true value. Our method learns a long-tailed distribution based on the training set data, dynamically adjusting the allocation ratio for novel classes. Compared to UNO, our method can increase the weight of the head class, which can suppress the decomposition of head classes into multiple subcategories. As a result, only a small number of categories are allocated to categories that do not match the ground truth, as shown the purple-bordered boxes in the right side image in Figure 8.

G Without strong model on CIFAR and ImageNet

We conduct experiment on CIFAR100 and ImageNet100 datasets using a non-strong model. We first perform unsupervised pre-training of the model on a long-tailed dataset using MoCoHe et al. (2020) on known classes. Next, we conduct supervised training on the known classes data. Finally, we discover the novel classes by jointly training on the known and novel classes.

The results in Tab.12 demonstrate that our method outperforms existing methods on the novel classes in both CIFAR100 and ImageNet100 datasets, especially in the more challenging setting where $R^u = 100$. However, on the CIFAR dataset, when equipped with the LA or cRT technique, the baseline method surpasses our

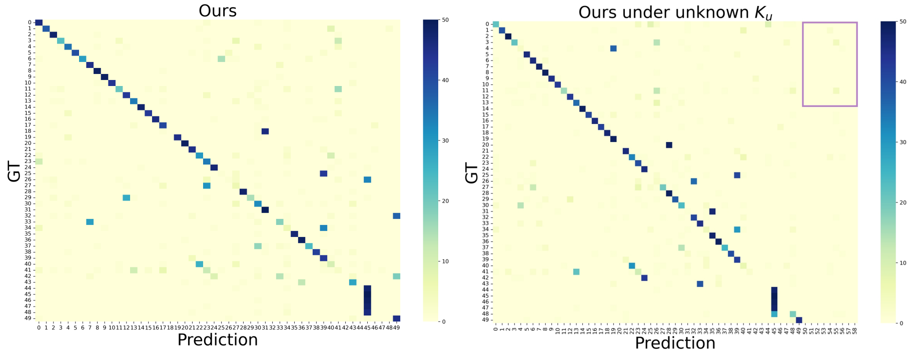

Figure 8: The confusion matrix of Ours for known and unknown K^u

Table 12: The performance on CIFAR-100, ImageNet100 without strong model.

	CIFAR100-50-50						**ImageNet100-50-50**					
	$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$			$R^s = 50, R^u = 50$			$R^s = 50, R^u = 100$		
Method	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**
AutoNovel	29.72	16.90	42.54	30.39	16.82	43.96	45.52	25.24	65.80	42.88	19.56	66.20
AutoNovel+LA	30.74	18.60	42.88	30.54	17.04	44.04	45.90	25.28	66.52	43.04	19.88	66.20
AutoNovel+cRT	30.72	18.38	43.06	29.35	16.20	42.50	46.84	27.20	66.48	42.78	21.20	64.36
UNO	33.80	27.04	40.56	33.05	24.64	41.46	43.52	26.88	60.16	42.00	24.88	59.12
UNO+LA	34.78	27.50	42.06	34.66	24.04	45.28	45.78	29.08	62.48	44.80	26.80	62.80
UNO+cRT	36.98	29.44	44.52	33.27	25.50	41.04	43.72	27.36	60.08	43.16	25.92	60.40
Ours	36.42	30.22	42.62	35.08	27.36	42.80	47.66	29.48	65.84	47.08	27.96	66.20
Ours+LA	37.51	30.72	44.30	35.84	27.50	44.18	48.90	29.28	68.52	48.04	27.80	68.28

method on the known classes, resulting in our method being slightly better or worse than existing methods in overall accuracy. While on ImageNet100 datasets, our method surpasses the existing method by a sizeable margin. Furthermore, when applying the LA technique to our method, our results consistently outperform the existing methods in terms of the "All" and "Novel" metrics.

H Known data are balanced

We conduct the experiment when known classes are balanced on CIFAR100 and ImageNet100. The results in Tab.13 show we achieve consistent and significant improvement on novel classes. For CIFAR100, our method achieves large improvements in different R^u settings, surpasses the previous SOTA method by 5.74% and 6.20% in novel accuracy. For ImageNet100, we achieve a significant improvement over the previous SOTA method by 13.68% and 14.48% in novel accuracy.

Table 13: Experiments on balanced known classes

	CIFAR100-50-50						**ImageNet100-50-50**					
	$R^s = 1, R^u = 50$			$R^s = 1, R^u = 100$			$R^s = 1, R^u = 50$			$R^s = 1, R^u = 100$		
Method	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**	**All**	**Novel**	**Known**
Autonovel	55.66	24.24	87.08	54.60	22.24	86.96	69.20	45.28	93.12	66.60	39.60	93.60
Autonovel + LA	56.20	24.80	87.70	55.11	22.46	87.76	69.00	44.72	93.28	66.54	39.48	93.60
AutoNovel + cRT	57.11	27.78	86.44	56.01	25.88	86.41	69.71	45.74	93.68	67.35	41.23	93.48
UNO	59.84	32.88	86.80	57.19	27.16	87.22	68.68	44.04	93.32	67.84	41.60	94.08
UNO + LA	59.87	32.92	86.82	58.07	28.90	87.24	68.82	44.28	93.36	68.00	41.60	94.40
UNO + cRT	60.35	34.38	86.32	57.74	28.46	87.02	69.74	45.88	93.60	67.66	40.88	94.44
Ours	63.19	40.12	86.26	60.55	35.10	86.00	76.50	59.56	93.44	74.86	56.08	93.64